

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЗАПОРІЗЬКА ПОЛІТЕХНІКА»

Факультет комп'ютерних наук та технологій
Кафедра комп'ютерних систем та мереж

Пояснювальна записка

до дипломного проекту (роботи)

магістра

(ступінь вищої освіти)

на тему **РОЗРОБКА СИСТЕМИ ПЛАНУВАННЯ ВИРОБНИЦТВА
ІНСТРУМЕНТУ ТА ОСНАЩЕННЯ НА БАЗІ ПРИКЛАДНОГО
МОДУЛЮ PS SAP ERP**

Виконав: студент 2 курсу, групи **КНТ-622м**
спеціальності _____

123 Комп'ютерна інженерія

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Спеціалізовані комп'ютерні системи

КОЛОЩУК Д.А.

(ПРИЗВИЩЕ та ініціали)

Керівник **КУДЕРМЕТОВ Р.К.**

(ПРИЗВИЩЕ та ініціали)

Рецензент **СТЕПАНЕНКО О.О.**

(ПРИЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет _____ Комп'ютерних наук і технологій
 Кафедра _____ «Комп'ютерні системи та мережі»
 Ступінь вищої освіти _____ магістерський
 Спеціальність _____ 123 Комп'ютерна інженерія
 (код і найменування)
 Освітня програма (спеціалізація) _____ Спеціалізовані комп'ютерні системи
 (назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Зав. кафедри Кудерметов Р.К.

« _____ » _____ 2023 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА

КОЛОЩУКА Дмитра Анатолійовича

(ПРИЗВИЩЕ, і'мя, по батькові)

1. Тема проєкту (роботи) Розробка системи планування виробництва інструменту
та оснащення на базі прикладного модулю PS SAP ERP
 керівник проєкту (роботи) к. т. н., доцент, КУДЕРМЕТОВ Равіль Камілович,
 (науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від « 24 » жовтня 2023 року № 400

2. Строк подання студентом проєкту (роботи) 10 грудня 2023 року
 3. Вихідні дані до проєкту (роботи) наявні ERP-системи та існуюче на підприємстві
програмне забезпечення планування виробництва

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно
розробити)

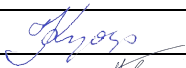
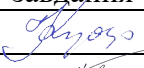
1) Аналіз технології ERP

2) Проектування системи

3) Розробка програмного забезпечення

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

6. Консультанти розділів проєкту (роботи)

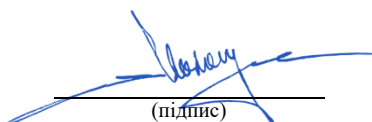
Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-3	КУДЕРМЕТОВ Р.К., к. т. н., доцент		
нормоконтроль	ЩЕРБАК Н.В., ст. викл.		

7. Дата видачі завдання 01.10.2023 року

КАЛЕНДАРНИЙ ПЛАН

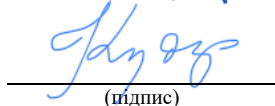
№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Аналіз технології ERP	20.10.2023	
2	Аналіз та вибір ERP-системи для розробки системи планування виробництва	31.10.2023	
3	Розробка архітектури системи планування виробництва	06.11.2023	
4	Розробка схеми системи планування виробництва	10.11.2023	
5	Розробка програмного забезпечення	20.11.2023	
6	Оформлення отриманих результатів у ПЗ	24.11.2023	
7	Оформлення допоміжного та графічного матеріалу	01.12.2023	

Студент


(підпис)

Дмитро КОЛОЩУК
(Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)


(підпис)

Равіль КУДЕРМЕТОВ
(Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

ПЗ: 82 с., 37 рис., 8 джерел.

АВАР/4, ERP-СИСТЕМА, МОДЕРНІЗАЦІЯ ТА ІНТЕГРАЦІЯ,
ПЛАНУВАННЯ ВИРОБНИЦТВА, РОЗРОБКА СИСТЕМИ.

Об'єкт дослідження – система планування допоміжного виробництва.

Мета дослідження – перенести функціонал програмного забезпечення для планування виробництва інструменту та оснащення з Clipper до системи ERP.

У першому розділі розглянуті основні можливості існуючих систем ERP, наведено функціонал та особливості, порівняні найбільш популярні системи у цій галузі.

У другому розділі проведено проектування системи, розглянуті аспекти системи планування виробництва, архітектура обраної ERP-системи, наведена схема системи планування допоміжного виробництва.

У третьому розділі наведено реалізацію спроектованої системи, результати та приклади роботи розроблених програм.

У результаті виконання проекту функціонал програмного забезпечення для планування виробництва інструменту та оснащення було перенесено з Clipper до системи ERP. Це дало змогу інтегрувати всі наявні процеси виробництва з загальновиробничою системою.

ABSTRACT

Explanatory note to the master's work: 82 p., 37 figures, 8 sources.

ABAP/4, ERP SYSTEM, MODERNIZATION AND INTEGRATION,
PRODUCTION PLANNING, SYSTEM DEVELOPMENT.

The object of research is the auxiliary production planning system.

The goal of the research is to transfer the functionality of the software for planning the production of tools and equipment from Clipper to the ERP system.

The first section discusses the main capabilities of existing ERP systems, providing functionality and features, and comparing the most popular systems in this field.

The second section involves system design, considering aspects of production planning, the architecture of the chosen ERP system, and presenting a diagram of the auxiliary production planning system.

The third section provides the implementation of the designed system, along with results and work examples of the developed programs.

As a result of the project's execution, the functionality of the software for planning the production of tools and equipment was transferred from Clipper to the ERP system. This allowed for the integration of all existing production processes with the overall production system.

ЗМІСТ

Скорочення та умовні позначки	8
Вступ.....	9
1 Аналіз технології ERP	11
2 Проектування системи.....	18
2.1 Основні поняття та задачі планування виробництва.....	18
2.2 Архітектура системи планування виробництва	20
2.3 Схема системи планування виробництва	22
2.4 Налаштування клієнта у системі	23
2.5 Диспетчер і робочі процеси системи.....	26
2.6 Апаратна та програмні частини ERP-системи	31
2.6.1 Апаратна частина	32
2.6.2 Програмна частина.....	33
3 Розробка системи	34
3.1 Таблиці баз даних.....	34
3.1.1 Створення таблиць баз даних у середовищі АВАР/4.....	34
3.1.2 Структурна схема бази даних	37
3.1.3 Опис необхідних таблиць	38
3.2 Розробка програмного забезпечення.....	39
3.3 Розробка інтерфейсів користувача	43
3.4 Опис програм	45
3.4.1 Програма створення картки ЗТО	45
3.4.2 Програма ведення виробничої бази.....	46
3.4.3 Програма формування плану виробництва	51

3.4.4 Приклади роботи програм.....	53
Висновки	62
Перелік джерел посилання	63
Додаток А.....	64
Додаток Б.....	69
Додаток В	76

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

ERP	– Enterprise Resource Planning
MRP	– Material Requirement Planning
MRP II	– Manufacturing Resource Planning
SAP	– Systemanalyse Programmentwicklung
CPM	– Cost per 1000 impressions
SCP	– Supply Chain Planning
PLM	– Product Lifecycle Management
CRM	– Customer Relationship Management
ABAP/4	– Advanced Business Application Programming
ЗТО	– засоби технологічного оснащення
SAPGUI	– Systemanalyse Programmentwicklung Graphical User Interface
CCMS	– Computing Center Management System
LUW	– Logical Unit of Work
CAD	– Computer Aided Desing
DYNPRO	– Dynemic Products

ВСТУП

Основним завданням керівництва виробничого підприємства є максимально ефективно використання наявних в розпорядженні виробничих, людських і фінансових ресурсів. Продукція має виготовлятися з мінімальними витратами та повністю відповідати попиту – задовольняти потреби клієнтів, вироблятися своєчасно і у достатній кількості. Вирішення всіх цих завдань відбувається за допомогою планування виробництва, яке дозволяє визначити цільові показники та дії для їх досягнення з урахуванням зовнішніх та внутрішніх факторів. Для підприємства загалом планування дозволяє досягти бажаного положення на ринку, яке визначається показниками обсягу продажів, прибутку та частки ринку. Для структурних підрозділів та окремих співробітників планування забезпечує узгодженість дій при вирішенні поставлених завдань.

Зміна завжди – це не лише можливість, але й виклик до нових досягнень. У сучасному світі, де швидкість та точність є критичними факторами успіху, управління виробництвом вимагає інноваційних рішень. Щоб краще відповідати цим вимогам виробництво потребує впровадження системи ERP.

Впровадження системи ERP на нашому підприємстві у свій час було не тільки важливим технологічним кроком, а й стало новою сторінкою у його розвитку. Наразі у основному виробництві впроваджено майже всі з існуючих модулів ERP-системи.

Актуальність проекту полягає в необхідності модернізації системи планування допоміжного виробництва, а саме планування виробництва інструменту та оснащення. Існуюча система на базі Clipper стає застарілою та недостатньо ефективною для сучасних потреб. Перенесення програмного забезпечення до системи ERP дозволить оптимізувати процеси виробництва, впровадивши нові можливості та забезпечивши більшу інтеграцію з вже існуючими на підприємстві модулями системи ERP.

Існуюча на сьогоднішній день система планування виробництва інструменту та оснащення виявляє суттєві недоліки, тому вона не відповідає вимогам сучасного виробництва. Її базовий фундамент на основі Clipper, безперечно, в свій час був інноваційним, але у змінному світі технологій став перешкодою для оптимального функціонування. Застосування вже застарілої системи уповільнює робочі процеси, обмежує можливості адаптації до нових вимог та технологій, тим самим утруднюючи ефективне виробництво інструменту та оснащення. Що стосується потреб підприємства, існуюча система не забезпечує необхідної інтеграції та гнучкості у вирішенні виробничих завдань.

Перенесення програмного забезпечення до системи ERP має вирішити ці проблеми. Це оновлення підвищить швидкість та точність процесів планування, дозволить впровадити нові функції, що сприятимуть оптимізації виробничих процесів. Основним плюсом буде підвищена інтеграція з вже існуючою системою ERP, що забезпечить гармонійну взаємодію між різними відділами підприємства та сприятиме більшій ефективності у керуванні.

Основною метою проекту є перенесення функціоналу програмного забезпечення для планування виробництва інструменту та оснащення з Clipper до системи ERP.

Для виконання поставленої мети необхідно вирішити цілий ряд завдань:

- оцінка всіх аспектів та функціоналу, що були реалізовані у програмному забезпеченні на базі Clipper та визначення, які з них потрібно перенести або модернізувати для використання у системі ERP;
- узгодження з потребами користувачів та збір вимог щодо нового функціоналу та інтеграції з системою ERP для визначення оптимального рішення;
- реалізація програмного забезпечення у системі ERP, його тестування та інтеграція з основною системою.

Виконання всіх цих завдань забезпечить необхідну інтеграцію системи планування виробництва оснащення у основну ERP-систему підприємства.

1 АНАЛІЗ ТЕХНОЛОГІЇ ERP

Наявність в сучасній компанії інформаційної системи, яка допомагає зробити внутрішні бізнес-процеси ефективними, знизити витрати на виробництво і зробити підприємство привабливішим для інвесторів, має важливе значення для конкурентоспроможності. Один із яскравих прикладів таких систем є корпоративні інформаційні системи, які використовують методологію планування ресурсів підприємства ERP.

ERP – це інтегрована система методів, процесів, технологій і ресурсів, яка включає в себе управління ланцюжком постачання, оптимізацію планування і розкладу, автоматизацію продажів, конфігурування товарів, планування ресурсів, інтелектуальний аналіз бізнесу, OLAP-технології, електронну комерцію та керування даними про виробництво [1]. Основна мета ERP-систем – це автоматизація пов'язаних процесів планування, обліку та управління в основних сферах діяльності компанії.

Історично концепція ERP стала розвитком простіших концепцій планування матеріальних потреб MRP (Material Requirement Planning) та планування виробничих ресурсів MRP II (Manufacturing Resource Planning).

Класичні ERP-системи забезпечують функціонування наступних задач:

- керування фінансами;
- керування виробництвом;
- керування формуванням та розподілом запасів;
- керування реалізацією та маркетингом;
- керування утриманням покупців;
- керування постачанням;
- керування проектами;
- керування сервісним обслуговуванням;
- керування процедурами забезпечення якості продукції.

Знання, вбудовані в ERP-системи, допомагають вирішити завдання корпоративної автоматизації і представляти роботу всіх функціональних підрозділів компанії як частину єдиної складної системи [2].

Існує декілька альтернативних систем ERP, що надають необхідну нам функціональність:

- SAP ERP пропонує широкий вибір рішень, які включають різноманітні модулі для фінансів, планування та управління виробництвом, управління персоналом та ін.;

- Oracle ERP Cloud пропонує хмарні рішення ERP, які включають модулі для фінансів, управління персоналом, виробництва та інших бізнес-процесів;

- Microsoft Dynamics 365 пропонує модулі для управління ресурсами підприємства, фінансами, продажами, обслуговуванням клієнтів та іншими функціями;

- Infor ERP пропонує різноманітні рішення ERP для різних галузей, таких як виробництво, дистрибуція, охорона здоров'я тощо;

- IFS Applications пропонує рішення ERP для різних галузей, включаючи виробництво, енергетику, будівництво та обслуговування;

- Openbravo надає відкрите програмне забезпечення для управління ресурсами підприємства з фокусом на роздрібній торгівлі та дистрибуції.

При виборі ERP-системи важливо враховувати специфіку підприємства та порівняти кілька варіантів таких систем для здійснення остаточного вибору.

Порівняння ERP-систем може бути здійснене за допомогою ряду критеріїв, які дозволяють оцінити їх ефективність і відповідність потребам конкретного підприємства. Ключовими критеріями є функціональність, гнучкість та налаштування, вартість, інтеграція з іншими системами, масштабованість, безпека даних та мобільність. Ці критерії можуть бути використані для оцінки різних ERP-систем. Розглянемо детально три найвідоміші ERP-системи: Oracle ERP Cloud, Microsoft Dynamics 365, SAP ERP.

Oracle ERP Cloud – це хмарна система планування ресурсів підприємства, що складається з взаємозалежних модульних рішень [3]. Цей програмний продукт

надає автоматизацію та оптимізацію широкого спектру завдань різних рівнів – від управління ризиками та проектами до корпоративного планування та бюджетування. Платформа включає декілька десятків бізнес-додатків для управління фінансами, закупівлями, замовленнями, ефективністю компанії (CRM), персоналом, послугами, взаємовідносинами з клієнтами, логістикою. Ця система адаптується під потреби бізнесу та розширюється за допомогою інших продуктів Oracle або інтегрованого стороннього програмного забезпечення.

Цей інструмент підходить для підприємств малого, середнього та великого бізнесу, яким потрібна масштабована інформаційна платформа для автоматизації всіх видів діяльності. Завдяки розділеному доступу, цей програмний продукт може використовуватися як фахівцями (бухгалтерами, керівниками складів, менеджерами), так і управлінцями.

Продукт збирається із самостійних SaaS-додатків за принципом конструктора, враховуючи специфіку діяльності підприємства. Під час впровадження додатки налаштовуються під потреби замовника, а при подальшому зростанні підприємства – масштабуються за допомогою додаткових модулів. У повному обсязі комплексна система охоплює всі життєві цикли компанії на будь-яких рівнях – від закупівель розхідних матеріалів для офісів до укладання договорів з зарубіжними партнерами та підведення фінансових підсумків. Цей інструмент забезпечує обробку даних будь-яких бізнес-процесів, на основі яких формуються звіти в різних розрізах. Серед всіх модулів основними є – ERM, планування (постачання, кадри, фінанси), бухгалтерський облік, управління (закупівлі, запаси, проекти, ресурси, фінанси).

Ключові особливості Oracle ERP Cloud:

- модульна структура адаптується під специфіку підприємства;
- єдина модель даних для всього підприємства;
- вбудована аналітика без сховища даних;
- вбудовані інтелектуальні засоби обробки зображень;
- хаб бухгалтерського обліку;
- інтеграція з Excel.

Модулі Oracle ERP Cloud відповідають за різні аспекти планування виробництва та управління бізнес-процесами. Модуль для оптимізації ланцюга постачання Supply Chain Planning (SCP) включає в себе можливості прогнозування попиту, управління запасами, планування оптимальних запасів і координацію з постачальниками. Модуль керування запасами Inventory Management дозволяє керувати запасами, включаючи ведення обліку матеріалів, контроль за рівнями запасів, оптимізацію управління запасами та зниження витрат. Модуль виробництва Manufacturing дозволяє керувати виробництвом, включаючи планування виробництва, управління операціями виробництва, контроль якості і виконання виробничих замовлень. Модуль управління Product Lifecycle Management (PLM) дозволяє керувати життєвим циклом продукту, включаючи проекти, специфікації, документацію і процеси зміни. Модуль централізованого планування Planning Central об'єднує різні аспекти планування, такі як попит, запаси, виробництво та постачання, для оптимізації процесів. Модуль попиту Demand Management дозволяє прогнозувати та управляти попитом на продукцію, аналізувати тенденції споживання, планувати продажі та прогнозувати попит на товари. Модуль управління замовленнями Order Management дозволяє керувати процесом замовлення від початку до кінця, включаючи прийом замовлення, виконання, доставку та обробку повернень. Ці модулі співпрацюють між собою, щоб забезпечити цілісне управління бізнес-процесами виробництва та плануванням у реальному часі. Кожен з них спеціалізується на конкретному аспекті, але їхні можливості можуть взаємодіяти для оптимізації управління виробництвом і постачанням. Процеси, за які відповідають ці модулі, можуть відрізнятися в залежності від конкретних вимог підприємства та конфігурації системи. Зазвичай Oracle ERP Cloud надає гнучкі налаштування для пристосування до різних бізнес-потреб.

Microsoft Dynamics 365 – це нове покоління інтелектуальних бізнес-додатків, які допомагають компаніям розвиватися та трансформуватися. В склад продукту входять нові спеціалізовані додатки з функціями CRM та ERP, а також відмінні можливості інтеграції [4].

Microsoft Dynamics CRM – це система для управління проектами та комунікацією з клієнтами. Представляє собою розширену платформу – доступ до ліцензійної системи здійснюється через браузер, мобільні телефони або плагін у Microsoft Outlook. Гнучка настройка дозволяє використовувати систему відповідно до унікальних цілей компанії. Програму можна використовувати на основі хмарних технологій, а також доступна локальна версія на сервері.

Microsoft Dynamics ERP – це сімейство програмних продуктів, розроблених для автоматизації бізнес-процесів підприємств різних розмірів і галузей. Ця система поєднує в собі різні модулі та рішення для управління фінансами, виробництвом, логістикою, ресурсами, продажами, кадрами, закупівлями та іншими аспектами діяльності компанії.

Ключові особливості Microsoft Dynamics 365:

- інтегрована та хмарна платформа;
- гнучка адаптація;
- єдина модель даних;
- вбудована аналітика;
- ефективність та автоматизація рутинних процесів;
- інтеграція з Excel.

Microsoft Dynamics 365 має різноманітні модулі. Модуль Supply Chain Management надає інструменти для оптимізації ланцюга постачання. Функціонал включає управління запасами, прогнозування попиту, планування виробництва, управління замовленнями та постачанням. Модуль Finance дозволяє управляти бюджетами, фінансовою звітністю, операціями з оплати та внутрішнім обліком, що є важливим елементом для планування та контролю витрат. Модуль Commerce дозволяє керувати роздрібними операціями, включаючи управління продажами, інвентаризацією та взаємодію з клієнтами, що може бути важливим для планування попиту та управління запасами. Модуль Project Operations включає планування ресурсів, управління витратами та виконанням проектів, що може бути важливим для планування виробництва і ресурсів. Модуль Field Service дозволяє планувати, виконувати та відстежувати обслуговування клієнтів на

місці, управляти ресурсами, планувати та виконувати завдання, що може бути важливим для оптимізації процесів виробництва. Ці модулі співпрацюють між собою, надаючи інтегровані рішення для планування, виробництва, управління запасами та інших бізнес-процесів.

SAP ERP-система має в своєму складі різні функціональні модулі, які виконують роль державної локалізації, спрямованої на виконання вимог чинного законодавства [5]. Ці модулі включають інтерактивні звіти, такі як оборотно-сальдова відомість в матеріальному обліку, друкарські форми, як рахунок-фактура, стандартні форми матеріального обліку (форми М-4 «Прибутковий ордер», М-11 «Накладна вимога на відпуск (внутрішнє переміщення) матеріалів», М-20 «Накладна на відпуск товарно-матеріальних цінностей» і інші), а також функціональні елементи діалогових транзакцій.

Функціональна модель SAP ERP складається з різних прикладних модулів, які підтримують різні бізнес-процеси підприємства і інтегровані в реальному часі. Серед ключових модулів можна виділити модуль фінансів FI, модуль контролінгу CO, управління основними коштами AM, управління проектами PS, виробниче планування PP, управління матеріальними потоками MM, збут SD, управління якістю QM, техобслуговування та ремонт обладнання PM та управління персоналом HR. Система логічно поділяється на різні модулі, кожен з яких містить безліч транзакцій, які охоплюють конкретний аспект діяльності підприємства. Межі між цими модулями умовні, і між ними відбувається обмін даними, можуть бути спільні налаштування та таблиці з даними, а також спільне використання частини програм на ABAP/4.

Базисна система є основою для SAP R/3 і забезпечує інтеграцію всіх прикладних модулів, незалежність від апаратної платформи та можливість роботи в розподіленій архітектурі клієнт-сервер. Система SAP R/3 може працювати на різних серверах та з різними системами управління базами даних. Користувачі можуть працювати в різних середовищах, таких як Windows, OSF/Motif, OS/2 або Macintosh. Базисна система є ключовим модулем, який забезпечує правильну роботу системи та відповідає за адміністрування та підтримку системи SAP R/3.

Для підвищення ефективності підприємства та забезпечення його конкурентоздатності необхідно перейти від традиційних функціонально орієнтованих структур до більш гнучких процес-орієнтованих форм організації. Однак цей перехід може бути здійснений лише в разі наявності необхідних інструментів. Для системи SAP ERP таким інструментом є спеціалізований інструмент бізнес-інженерії Business Engineer, який дозволяє конфігурувати та налаштовувати SAP ERP так, щоб вона задовольняла потреби конкретного підприємства та підтримувала цю відповідність протягом усього життєвого циклу системи. Business Engineer має відкритий стандартний інтерфейс, який дозволяє партнерам SAP і консультантам створювати галузеві рішення на основі бізнес-сценаріїв SAP ERP та розробляти власні шаблони для впровадження системи SAP ERP.

Враховуючи ретельне порівняння трьох систем ERP – Oracle ERP Cloud, Microsoft Dynamics 365 та SAP ERP, для розробки нової системи планування допоміжного виробництва обираємо систему SAP ERP. Це рішення ґрунтується на кількох ключових факторах, які відображають потреби та стратегію розвитку нашого підприємства.

Перш за все, SAP ERP вже впроваджено у нашому підприємстві, що створює певні переваги в плані стабільності та знайомства наших працівників з системою. Це важливо для забезпечення плавного переходу на новий функціонал і забезпечення мінімальних перерв у роботі.

Крім того, SAP ERP має широкий функціонал та ще більше можливостей для розширення та додаткового розвитку. Це робить його ідеальним кандидатом для розробки нових модулів та функціоналу для планування, оскільки вже існуюча база інтеграції спрощує процес створення нових інструментів та їх розгортання в середовищі, з яким ми вже працюємо.

Приймаючи до уваги ці фактори, вважаю, що вибір SAP ERP – це найефективніший шлях для нашого підприємства, оскільки він дозволить нам максимально використати вже існуючу інфраструктуру та легко розширювати її для майбутніх потреб у плануванні та розвитку.

Зважаючи на основну мету проекту по перенесенню функціоналу програмного забезпечення для планування допоміжного виробництва з Clipper до системи ERP, слід зробити наступні уточнення щодо розробки:

- функціонал, реалізований у програмному забезпеченні на базі Clipper буде модернізований за рахунок використання нової архітектури системи ERP;
- потреби користувачів та новий функціонал інтерфейсів буде оптимально поліпшений для використання;
- розроблювана система буде забезпечувати ведення бази первинних даних для планування виробництва, відстеження етапів проходження замовлень з автоматичним заповненням даних руху щодо стану у виробництві, автоматичне формування планів виробництва з заданими параметрами;
- реалізація програмного забезпечення та його тестування, інтеграція з основною системою буде виконуватись за рахунок прикладних модулів системи.

2 ПРОЕКТУВАННЯ СИСТЕМИ

2.1 Основні поняття та задачі планування виробництва

Роботу допоміжного виробництва можна охарактеризувати такими роботами як, накопичення інформації згідно існуючого портфелю замовлень, його аналіз на відповідність потужності виробництва та контроль термінів виконання замовлень, укрупнене планування виробництва інструменту та оснащення (відстеження етапів виробництва), оперативне планування виробництва інструменту та оснащення на основі маршруту виготовлення у виробничих групах підрозділів.

На теперішній час планово-управлінські функції допоміжного виробництва повністю автоматизовані на Clipper. Програмне забезпечення було розроблене з урахуванням існуючого на той час обчислювального обладнання та можливостей програмування, тому воно складається з окремих задач, у зв'язку з чим частина

інформації дублюється, що знижує продуктивність та ефективність її використання. Задачі, які реалізовані на Clipper замикались тільки усередині допоміжного виробництва, а обмін між службами виконувався на паперових носіях. З оновленням та розширенням підприємства, впровадженням нових систем управління, виникла потреба у переході на більш інноваційні системи планування. На протязі періоду з 2012 по 2023 роки, 35% програмного забезпечення було перенесено з Clipper у відповідні модулі системи SAP. Фактично, це програмне забезпечення становить основу для розроблюваної системи планування виробництва у частині встановлення пріоритетів виготовлення інструменту та оснащення.

За причиною специфіки виробництва, використання стандартних модулів для розробки виробничого планування стало неможливим. За прикладом та логікою програмного забезпечення, написаного на мові Clipper, у проекті будуть реалізовуватись три основні програми для безпосереднього планування у підрозділах допоміжного виробництва. Дані, які раніше були внесені до програми у форматі *.dbf, перенесуться у відповідні таблиці системи SAP R/3. Розроблені програми будуть доповнювати існуючий каталог інструменту та оснащення, базу заявок та замовлень, архів виконаних замовлень и створювати базу анульованих записів. Останнє не було реалізовано на Clipper.

Безпосередньо плануванням виробництва інструменту та оснащення займаються три окремих підрозділи. В кожному із підрозділів, в залежності від специфіки виробництва, використовується певне програмне забезпечення на Clipper. Система, яка розробляється, буде поєднувати у собі функціонал необхідний цим підрозділам.

Задачею допоміжного виробництва є забезпечення підрозділів основного виробництва інструментом та оснащенням.

Інструмент та оснащення, які виготовляються на підприємстві, є засобами технологічного оснащення (ЗТО). Для ідентифікації ЗТО на підприємстві використовується спеціальна система кодування.

В залежності від методу застосування, ЗТО поділяються на типи. Виготовлення різних типів ЗТО закріплене за різними групами підрозділів в залежності від специфіки виробництва.

Кожна одиниця ЗТО має різну складність виготовлення, яка вимірюється нормо-годинами та має назву трудомісткість виробництва. Слід зазначити, що деякі ЗТО мають таку трудомісткість, що термін виготовлення може займати декілька місяців. Підрозділи, які виготовляють ЗТО мають обмеження по об'єму робіт, які вони можуть виконати протягом місяця. Цей об'єм вимірюються у нормо-годинах та має назву потужність підрозділу.

Найважливішим принципом планування є формування плану виробництва таким чином, щоб об'єм ЗТО, яких необхідно виготовити протягом місяця, не перевищував потужність підрозділу, а також забезпечував виготовлення ЗТО у вказаний термін.

Підставою для включення коду ЗТО у план виробництва є заявка від підрозділу замовника. Після включення коду ЗТО у план виробництва створюється замовлення на виготовлення.

Слід зазначити, що на формування плану виробництва суттєво впливають наступні фактори: пріоритетність тих чи інших виробів основного виробництва, потреба в ЗТО, виготовлення яких регламентується внутрішніми організаційно-розпорядчими документами, забезпечення ЗТО згідно встановлених норм витрат.

2.2 Архітектура системи планування виробництва

Архітектура системи «клієнт-сервер» розділена на три рівні [6]. Клієнт-серверні обчислення забезпечують величезні переваги щодо розподілу навантаження на систему, масштабування та гнучкості, необхідної для розвитку.

Тришарова архітектура системи складається з рівня презентації, рівня додатків та рівня баз даних – це оптимальна реалізація режиму клієнт-сервер. Характеристики кожного рівня:

- рівень презентації керує діалогом між кінцевим користувачем і яким-небудь застосунком, програмою;
- рівень додатків здійснює трансформацію даних, в чому, по суті, і полягає призначення додатків;
- рівень баз даних здійснює зберігання, оновлення та надання даних за допомогою програм, розташованих на рівні додатків.

За останнє десятиліття тришарова архітектура стала дуже популярною через значну гнучкість, яку надає цей підхід до обробки даних. Додатки можуть встановлюватися, конфігуруватися та функціонувати на одному центральному комп'ютері або розподілятися між кількома комп'ютерами в залежності від типу додатків та навантаження. Більше того, при зміні навантаження та вимог до обробки даних, відповідні системи можна вибірково модернізувати. Такий підхід до роботи системи призводить до значного покращення характеристик ефективності за незначних витрат та збереження вже інвестованих коштів у комп'ютерне обладнання. У системі SAP R/3 принцип «клієнт-сервер» служить основою спільної роботи несумісних компонентів програмного забезпечення, які існують у межах централізованої або розподіленої конфігурації на утворюючих мережу серверах. Обмін інформацією між серверами може відбуватися як за допомогою синхронного спілкування від програми до програми, так і шляхом асинхронного обміну повідомленнями або віддаленого використання SQL. Для окремих областей роботи також можуть встановлюватися спеціальні сервери додатків. Призначення окремих серверів дозволяє збалансувати навантаження, а також конфігурувати окремий сервер залежно від функцій, які він виконує. Таким чином, досягається не тільки підвищення пропускної спроможності системи та оперативності її реакції, але й оптимізація витрат.

Архітектура системи, яка розроблюється, представлена на рис. 2.1.

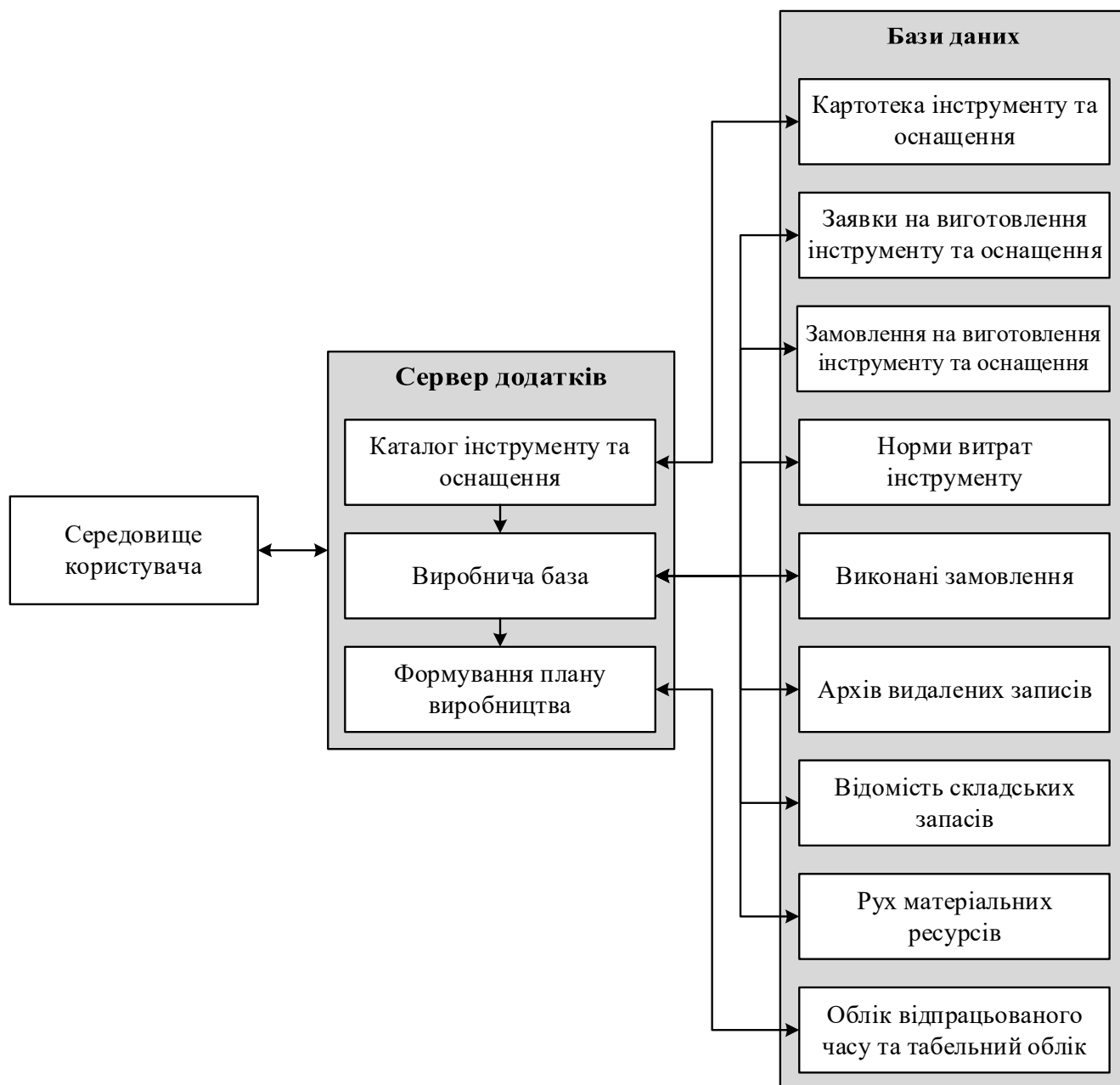


Рисунок 2.1 – Архітектура системи планування виробництва

2.3 Схеми системи планування виробництва

Для реалізації системи планування виробництва інструменту та оснащення необхідно створити:

- програму створення картки ЗТО у каталозі;

– програму ведення виробничої бази заявок та замовлень на виготовлення ЗТО;

– програму формування плану виробництва.

Всі вище вказані елементи повинні бути пов'язані з модулем HR у частині потужності виробництва та модулем MM у частині контролю запасів та руху матеріальних ресурсів.

Схема розроблюваної системи показана на рис. 2.2.

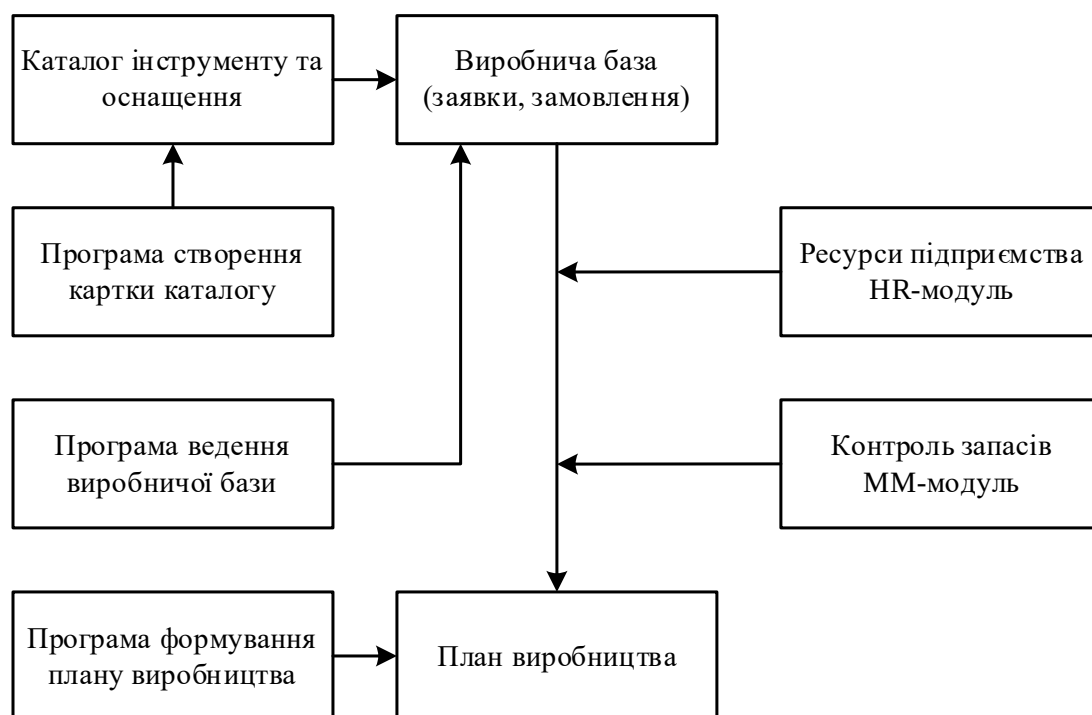


Рисунок 2.2 – Схема системи планування виробництва

2.4 Налаштування клієнта у системі

Для реалізації пунктів безпеки та захисту інформації у системі SAP передбачені засоби для визначення атрибутів, які надають клієнту конкретні характеристики (параметри), включаючи можливість або неможливість внесення змін. Налаштування опцій зміни системи визначають доступні функції налаштування та розробки в системі SAP. Ці налаштування є основою для

налаштувань всіх клієнтів системи (рис. 2.3). Іншими словами дозвіл на використання вищевказаних розроблюваних програм буде наданий окремим зареєстрованим користувачам системи, які мають право на використання транзакції, яка відповідає за планування допоміжного виробництва.

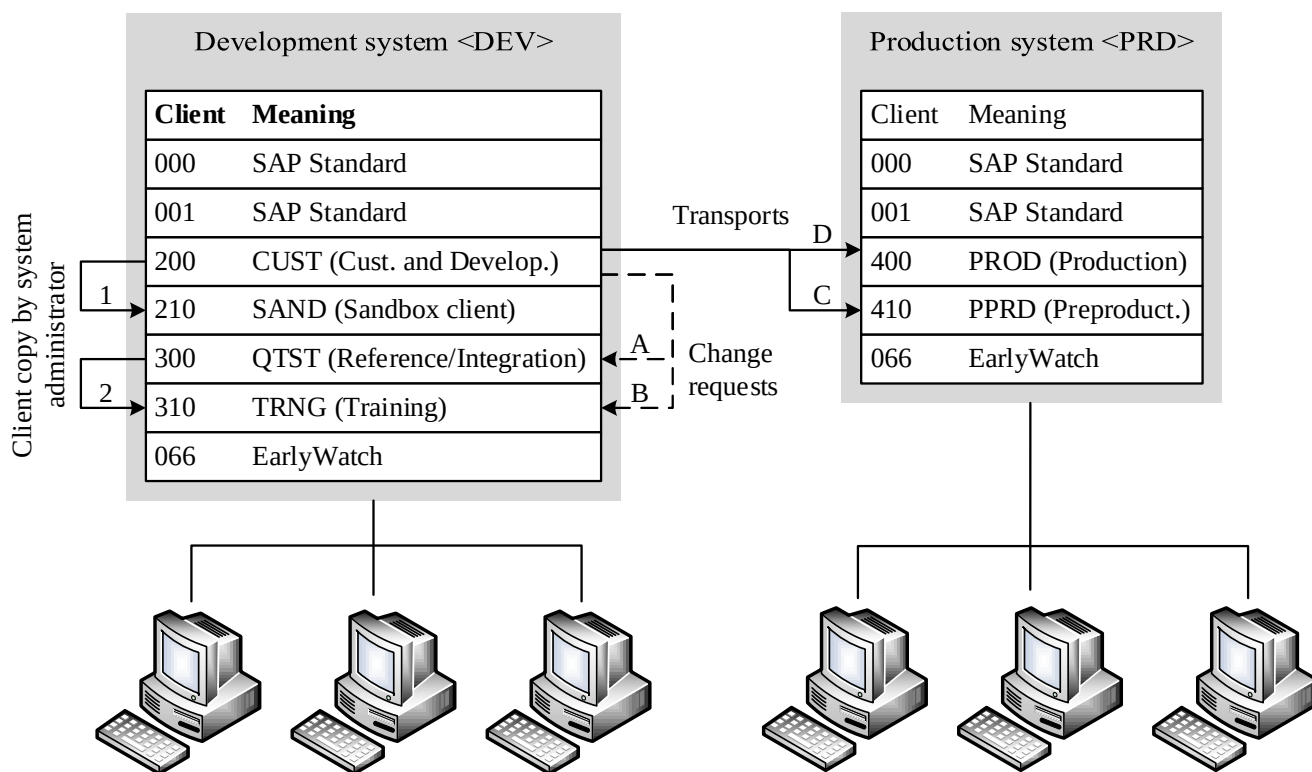


Рисунок 2.3 – Налаштування клієнтів шляхом копіювання клієнта або транспорту в двосистемному ландшафті

Клієнт – це самостійна організаційна одиниця в системі SAP R/3. У межах окремої системи SAP R/3 кожен клієнт ідентифікується тризначним номером. Однак однакові клієнти на кількох системах повинні ідентифікуватися однаковими номерами, ця міра необхідна для транспортування змінених об’єктів, оскільки з однієї системи змінений об’єкт за замовчуванням транспортується до такого самого Клієнта системи призначення. У кожного клієнта є своє середовище даних, включаючи налаштування (як загальні, так і специфічні для конкретного клієнта), об’єкти сховища, дані додатків (основні та за транзакціями) та основні записи користувачів.

Залежно від призначення, клієнти можуть бути одного з наступних типів:

- клієнт розробки;
- клієнт тестування;
- клієнт перевірки якості;
- клієнт для навчання;
- робочий клієнт;
- тестовий мандант (Sandbox);
- підготовка до виробництва.

Для розуміння принципу розмежування доступу та функцій зробимо опис налаштування клієнта в системі розробки DEV (двосистемний ландшафт).

000: Стандартний клієнт SAP – використовується для створення нових «нейтральних» клієнтів шляхом копіювання клієнта. Цей клієнт заборонено змінювати або видаляти.

001: Стандартний клієнт SAP – використовується для створення нових «нейтральних» клієнтів шляхом копіювання. Цей клієнт заборонено змінювати або видаляти.

200: CUST: Клієнт налаштувань та розробки – використовується для розробки програм. Усі зміни автоматично записуються та зберігаються в запитах на зміни. Перехресні налаштування для всіх клієнтів дозволені лише в цьому клієнті. У цьому клієнті заборонено створювати жодних тестових даних.

210: SAND: Клієнт пісочниці – використовується для ознайомлення з системою, випробовування додатків або тестування функцій налаштування. В цьому клієнті можливо виконувати налаштування, які специфічні для клієнта, зміни не реєструються. Будь-які зміни, які вносяться в цьому клієнті і повинні бути збережені в системі, необхідно повторно ввести вручну в клієнті CUST (200). Цей клієнт може бути оновлений адміністратором шляхом копіювання клієнта з CUST (200).

300: QTST: Довідковий клієнт для інтеграції та навчання – використовується для перевірки змін в налаштуванні та розробці перед інтеграційним тестуванням, а також для створення навчального клієнта TRNG (310) перед навчанням, щоб забезпечити однакові умови для всіх навчальних

сесій. У цьому клієнті заборонено вносити зміни в об'єкти сховища або виконувати перехресне налаштування. Функція для зміни об'єктів, специфічних для клієнта, також вимкнена.

310: TRNG: Навчальний клієнт – використовується для навчання кінцевих користувачів. У цьому клієнті містяться основні та транзакційні дані, щоб забезпечити реалістичне навчання. Розробка та налаштування, специфічні для клієнта, не дозволені, функція для зміни об'єктів також вимкнена.

Налаштування клієнта визначають налаштування та функції розробки в системі SAP. Ці налаштування не перевизначають опції зміни системи. Кожна система має визначену роль у системному ландшафті і налаштування для кожної ролі. Кожен клієнт в системі також має свої власні налаштування.

2.5 Диспетчер і робочі процеси системи

Виконавча система R/3 слідує за багатьма функціями, такими як управління пам'яттю, управління повноваженнями та ін., тобто те, що зазвичай здійснює операційна система. Але система SAP сама здійснює ці функції, щоб забезпечити мобільність та контроль над продуктивністю системи. Виконавча система R/3 написана на мові ABAP/4.

Сервер застосунків складається з диспетчера та кількох робочих процесів (рис. 2.4). Виконавча система R/3 працює як група співпрацюючих паралельних процесів. Диспетчер – це процес, який виступає у ролі системи моніторингу транзакцій. Він призначає завдання для інших процесів, координує та контролює їх виконання. Кожен режим роботи в SAP має свого диспетчера, який, в свою чергу, може управляти багатьма робочими процесами. Диспетчер забезпечує оптимізацію та збалансованість навантаження системи, причому збалансованість навантаження підтримується за допомогою обробника запитів, який отримує запит на доступ від презентаційного компонента.

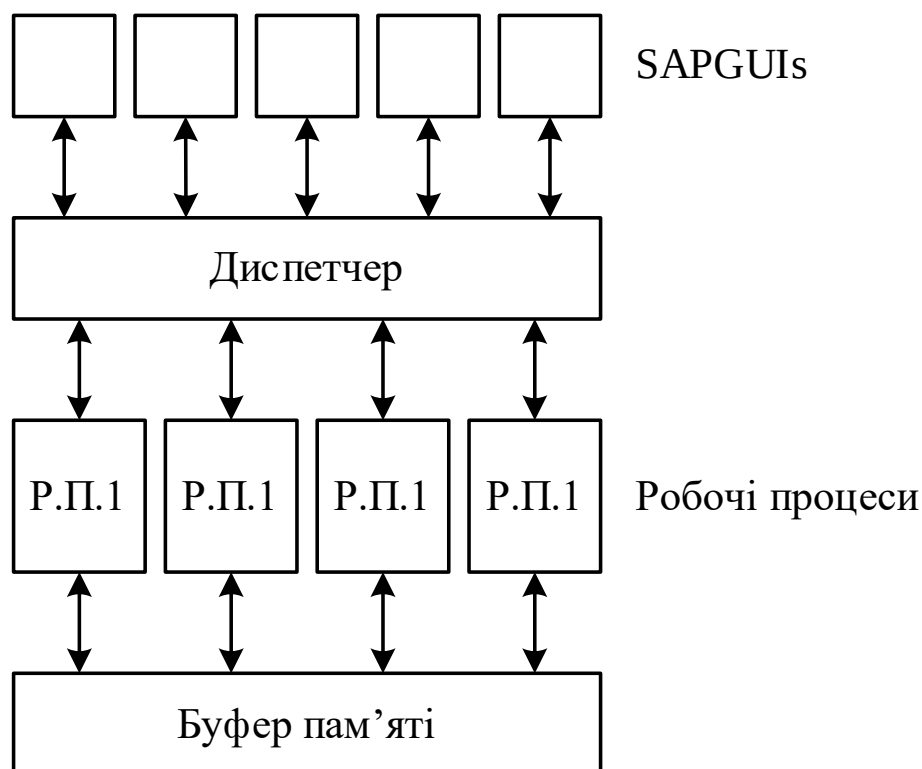


Рисунок 2.4 – Диспетчер і робочі процеси системи

Диспетчер оптимізує взаємодію між рівнем презентацій та рівнем додатків. Після отримання даних, введених користувачем, диспетчер ставить цей запит у чергу на обробку та призначає для цього вільний на даний момент робочий процес відповідного типу, причому спочатку обробляються запити, які надійшли раніше. Після завершення обробки даних результати надсилаються на відповідний екран, а робочий процес звільняється для обробки наступного запиту в черзі. Вся комунікація між робочими процесами та іншими системами, такими як графічний інтерфейс SAPGUI, а також комунікація між програмами, здійснюється через диспетчер. Кожен робочий процес складається з обробника завдань, який за вимогою активує процесор АВАР/4, інтерпретатор діалогу або інтерфейс для баз даних. Він також відповідає за завантаження та очищення інформації про користувача на початку та в кінці кожної стадії діалогу. Цю дію називають «розгортанням» та «згортанням» відповідно. Коли робочому процесу потрібні дані, що знаходяться на різних стадіях діалогу з користувачем, ці дані зберігаються в кеші – області пам'яті, доступній для різних програм.

SAPGUI – це стандартний графічний інтерфейс користувача системи R/3. Принцип роботи та визначення компонентів SAPGUI не пов'язані з системою представлення, тому інтерфейси користувачів SAP виглядають та функціонують однаково, незалежно від програм представлення, використаних для конкретної інсталяції. Графічні системи можуть базуватися на будь-якій платформі. Інтерфейс SAPGUI включає всі графічні можливості сучасних інтерфейсів – меню, панелі завдань, кнопки, перемикачі, електронний довідник, вказані списки значень полів тощо. Більше того, система SAP не передає повні графічні зображення екранів між своїми додатками, тому обсяг циркулюючих даних мінімальний та при одноразовій передачі зазвичай не перевищує 1-2 Кб. У результаті навантаження на мережу значно зменшується, що вносить значний внесок у можливості масштабування систем SAP.

Відкритий характер архітектури SAP забезпечує гнучкість системи, що дозволяє працювати одночасно з різними функціями та рішеннями на всіх рівнях, включаючи:

- рівень графічного інтерфейсу;
- рівень робочої станції;
- рівень додатків;
- рівень баз даних;
- рівень зовнішніх інтерфейсів;
- рівень протоколів передачі даних;
- рівень обладнання та операційних систем.

Вибір процесів передбачає систематичну компіляцію всіх процесів та варіантів процесів, що переважають на підприємстві. Ці процеси можуть належати будь-якому організаційному рівню, незалежно від того, чи є вони комп'ютеризованими чи ні. Складання карти процесів підприємства передбачає докладне відстеження та відбір процесів, найбільш важливих для підприємства. Це передбачає необхідність уточнення докладної інформації про кожен процес: назву, призначення, відповідального за процес співробітника, опис процесу (включаючи опис типу вихідних даних та результату процесу), якість та

ефективність, підпроцеси, інтерфейси взаємодії з іншими функціями та системами, статус процесу (чи є процес винятковим), можливості для удосконалення, аналіз впливу припущених сценаріїв та інше. Скомпільована карта процесів буде основою під час підготовки функціонального тестування. Тестові сценарії пишуться на основі цих карт.

Навантажувальне тестування – це процес, схожий з функціональним тестуванням, який проводиться для традиційних систем. Це тестування передбачає оцінку очікуваного навантаження на систему у реальному середовищі та відтворення (симуляцію) цих умов у тестовому середовищі з метою перевірки, чи зможе система ефективно впоратися з очікуваним навантаженням в умовах запропонованої конфігурації. Це тестування також може включати оцінку обсягів даних, пропускну здатності мереж LAN/WAN, а також оцінку обслуговування центральними серверами інтерактивних операцій між серверами та терміналами.

Computing Center Management System (CCMS) – це зручний у використанні централізований інструмент системного адміністрування та управління R/3, орієнтований на графічне представлення. Його можливості включають інструменти для виконання системних операцій, контролю, моніторингу та оптимізації. Досягнення цих цілей можливо завдяки повній інтеграції CCMS не лише з системою R/3, але також з операційною системою, управлінням мережею та системою управління базами даних.

Всі сервери в системі SAP ведуть журнали повного обліку подій, що відбуваються в системі. В системі передбачена можливість надання детальної звітності за журналами залежно від обраних параметрів, таких як назва режиму SAP, ID користувача, код журналу обліку проблем, код транзакції або навіть конкретний робочий процес чи служба.

Системні журнали поділяються на наступні типи:

- локальний системний журнал – призначений лише для локального сервера;
- віддалений системний журнал – призначений для конкретного віддаленого сервера;

– всі віддалені системні журнали – призначені для всіх віддалених серверів і режимів;

– центральний системний журнал – призначений для всіх серверів.

Файли локального системного журналу оновлюються циклічно, на неперервній основі, у той час як центральні системні журнали конфігуруються з метою розділення на поточні та старі файли. Журнали оновлюються у поточних файлах, поки ці файли не заповняться, після чого поточні файли копіюються на місце віддалених старих файлів, після чого створюється новий файл для запису поточних записів журналів.

Пакетний процес корисний при роботі з програмами, що обробляють дані протягом тривалого часу. Таким програмам можна призначити певний час для роботи – вдень або вночі. У фоновій обробці даних диспетчер призначає один пакетний процес для одного фонових завдання, який виконує завдання до кінця. Це різко відрізняється від діалогової обробки даних, коли диспетчер кожного разу призначає нову LUW для наступного вільного діалогового процесу.

Для обробки пакетних або фонових завдань зазвичай використовуються ті ж програми, що і для синхронної обробки даних у режимі on-line. Центральна управляюча система (CCMS) керує пакетними процесами та призначає час для їх виконання.

Транзакція SAP може додати або змінити дані в базі даних синхронно, тобто безпосередньо, в режимі реального часу, або асинхронно, тобто не безпосередньо.

При прямому запису даних в базу даних цю операцію виконує діалоговий робочий процес. При асинхронному записі фаза on-line (діалог з користувачем) та фаза оновлення даних виконуються окремо. Діалоговий процес створює окремий запис у журналі для оновлення даних одразу після завершення фази on-line. Це вигідно, оскільки під час діалогу дозволяє одразу переходити до наступного етапу діалогу, що заощаджує час і помітно підвищує ефективність роботи з системою. Тим часом, основний компонент запису стосується оновлень, пов'язаних зі змінами ділових операцій, які відбуваються в реальному часі (наприклад, розміщення запасів, отримання товарів та ін.), вторинний компонент стосується

оновлень, більш пов'язаних з бізнес-орієнтованими вимірами характеристик роботи за певний період. Отже, основні компоненти складаються з пріоритетних оновлень і повинні виконуватися якомога швидше. Оновлення основних компонентів повинно завершуватися до початку оновлення вторинних компонентів у базі даних.

Процеси, що протікають на серверах додатків, використовують службу повідомлень для обміну даними та інформацією. Служба шлюзів забезпечує комунікацію між системою R/3 та іншими зовнішніми системами. У основному ця служба займається передачею великих обсягів, пов'язаних з додатками даних. Для передачі інформації ця служба використовує протокол CPI-C.

2.6 Апаратна та програмні частини ERP-системи

Дуже важливе питання при впровадженні ERP-системи – тип і потужність передбаченого комп'ютерного обладнання, що складає інфраструктуру системи – має велике значення для можливості проведення різних операцій в масштабі всього підприємства.

Комп'ютерне обладнання є головним рушійним механізмом, необхідним для функціонування ERP в масштабі всього підприємства. Природа ERP-систем вимагає або монолітного сервера, або набору серверів, кожен з яких спеціалізувався би на певній частині загального передбаченого навантаження на систему. Кількість комп'ютерів у такому наборі може відповідати або функціональним областям, або кількості підрозділів, які система буде обслуговувати. Розміри та потужність комп'ютерів у цьому наборі залежать від передбаченого навантаження на систему та кількості терміналів, які будуть підключені до системи.

Рішення щодо системного програмного забезпечення приймаються одночасно з придбанням обладнання. Операційне середовище системи повинно

забезпечувати незалежність операцій від збоїв, а також засоби для автоматичного створення резервних копій даних, реплікацій і відображення даних в масштабі всієї системи, а також засоби автоматичного відновлення втрачених або пошкоджених даних.

Так як ERP-системи орієнтовані на кінцевого користувача, навантаження на систему зазвичай передбачається значне. Мережа повинна мати високу пропускну здатність, щоб не створювати перешкод у роботі, що ведеться з терміналів.

Відкриті системні інтерфейси дозволяють інтегрувати SAP з іншими програмами за допомогою стандартних інтерфейсів, таких як Object Linking Enabling (OLE) та Remote Function Call (RFC). Крім того, система R/3 відповідає наступним міжнародним стандартам:

- CPI-C для обміну даними між програмами на різних комп'ютерах;
- SQL та ODBC для доступу до баз даних;
- протокол передачі даних TCP/IP;
- RFC для високотехнологічних програмних інтерфейсів;
- OLE/DDE та RFC для інтеграції з програмами на основі ПК;
- x.400/x.500 як відкритий інтерфейс електронної пошти;
- протокол EEDIFACT EDI для обміну структурними бізнес-даними на рівні додатків;
- ALE для інтеграції децентралізованих додатків на різних комп'ютерах в режимі on-line;
- відкриті інтерфейси для взаємодії з системами, такими як системи контролю процесів, системи збору робочих даних, CAD та ін.

2.6.1 Апаратна частина

Сервер бази даних має бути потужним, з підтримкою високої продуктивності та місткості для зберігання даних. Операційна система серверу – SAP HANA або Oracle Database.

Сервер застосунків повинен забезпечувати стабільну роботу модуля SAP ERP PS та швидку реакцію системи. Операційна система – SAP NetWeaver або аналогічна для підтримки SAP ERP.

Сховище даних повинно забезпечувати високу продуктивність та надійність для зберігання планів, даних проектів та ресурсів. Сховище даних – SAP HANA.

Мережева інфраструктура повинна включати в себе високошвидкісні мережеві пристрої для швидкого передавання даних між серверами та клієнтськими пристроями, механізми безпечного зв'язку для захисту конфіденційної інформації.

Розділення мереж буде виконуватись фізично за рахунок використання різних сегментів VLAN для серверів бази даних та серверів додатків з оптимізацією мережевої інфраструктури.

Розглянемо налаштування мережевої інфраструктури.

Оптимізація. Виділена пропускна здатність для окремої мережі між серверами – 10 Gbps Ethernet. Програмне забезпечення – комутатори з підтримкою високих швидкостей та обробки даних Cisco Catalyst 9000 Series.

Безпека. Мережеві брандмауери для сегментації мереж та VPN для захисту комунікацій. Програмне забезпечення – Cisco ASA (Adaptive Security Appliance).

Оптимізація SAP ERP. Конфігурація SAP NetWeaver для використання конкретних IP-адрес окремої мережі. Програмне забезпечення – SAP NetWeaver.

Тестування продуктивності швидкості передачі даних з використанням програмного забезпечення iperf.

Моніторинг мережі для відстеження мережевої активності та проблем з продуктивністю з використанням програмного забезпечення PRTG Network Monitor, SolarWinds Network Performance Monitor.

2.6.2 Програмна частина

Програмна частина включає в себе модуль управління проектами (PS), основні функції для створення, управління та відстеження проектів, а також виділення ресурсів та планування бюджету, має інтеграцію з іншими модулями управління обліком, матеріальними потоками та управлінням людськими ресурсами.

У ролі інтеграційних модулів використані інтерфейси для взаємодії з зовнішніми системами управління проектами. Використані стандарти обміну

даними такі як XML, JSON та ін. API та засоби інтеграції для зв'язку з системами, що керують портфелем проектів або іншими суміжними системами.

3 РОЗРОБКА СИСТЕМИ

3.1 Таблиці баз даних

3.1.1 Створення таблиць баз даних у середовищі АВАР/4

Усі об'єкти АВАР-розробника, такі як моделі даних та процесів, словник АВАР/4, звіти, динамічні програми, бібліотеки функціональних процесів, об'єкти авторизації та ін. записані у сховищі АВАР/4. Для створення таблиць та інших важливих елементів використовуємо цей інструмент [7]. Початковий екран АВАР-словнику представлений на рис. 3.1.

Словник АВАР/4 є логічним поданням даних, які записані у стандартних фізичних базах даних, таких як Oracle, DB2, Informix та ін.

У словнику вводимо назву таблиці, яку будемо створювати. Усі таблиці містять поля. Атрибути індивідуальних полів описуються як елементи даних та відповідні домени. Для таблиць задаються таблиці індексів, які спрямовані на зменшення часу доступу при операціях з таблицями. Усі таблиці, індекси бази даних, види та ін. можуть бути створені у словнику. Структури схожі на таблиці, тільки не містять ніяких даних. Вони відносяться до послідовності полів у словнику. Структурні об'єкти можуть використовуватися для визначення даних у інтерфейсах при роботі у групі модулів та екранів. Структури існують лише у словнику, у основних базах даних вони відсутні. Приклад структури таблиці наведено на рисунку 3.2.

Після створення необхідної таблиці можна переглянути у графічному вигляді відношення таблиці до інших таблиць по зовнішньому ключу (рис. 3.3). Для здійснення навігації використовуємо браузер репозитарію (рис. 3.4).

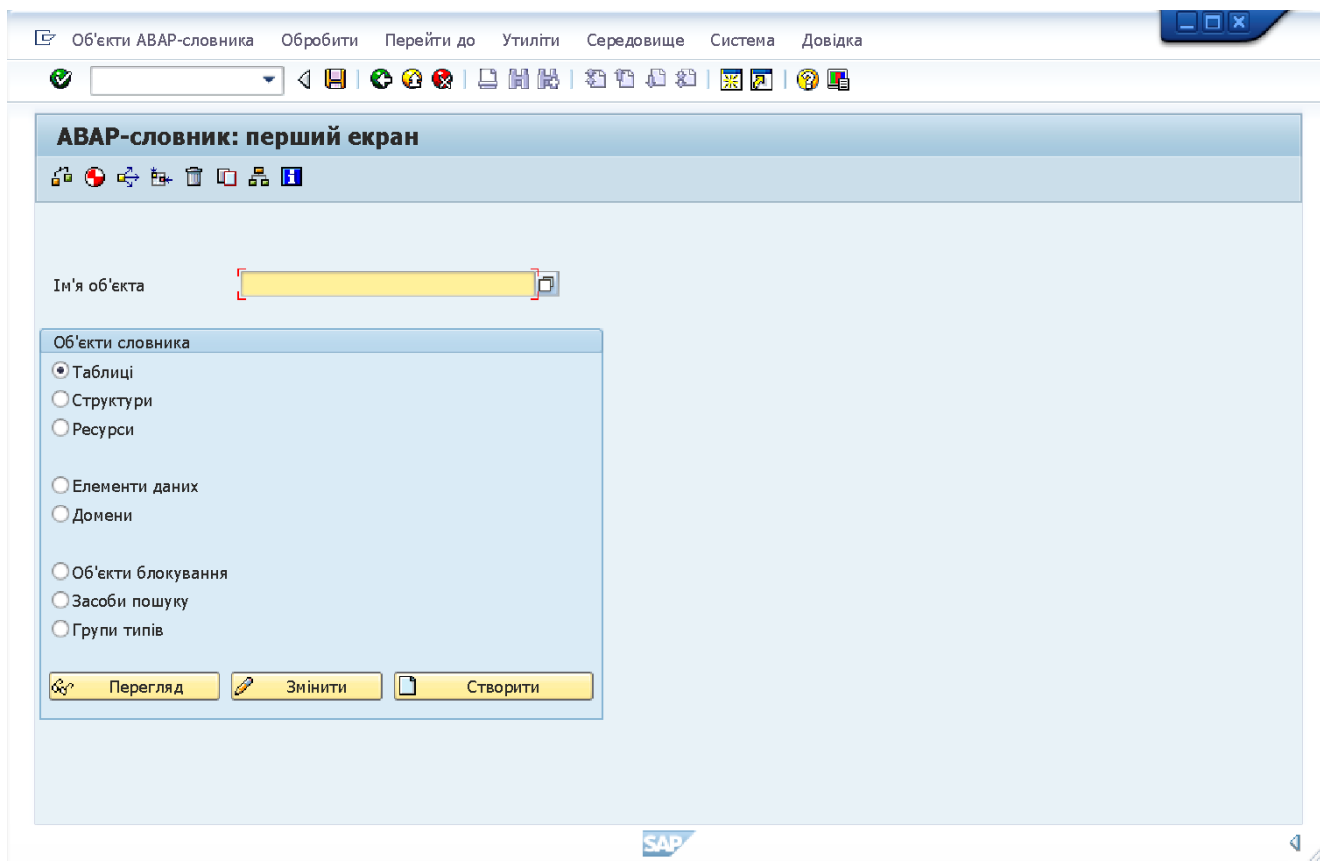


Рисунок 3.1 – Початковий екран АВАР-словнику

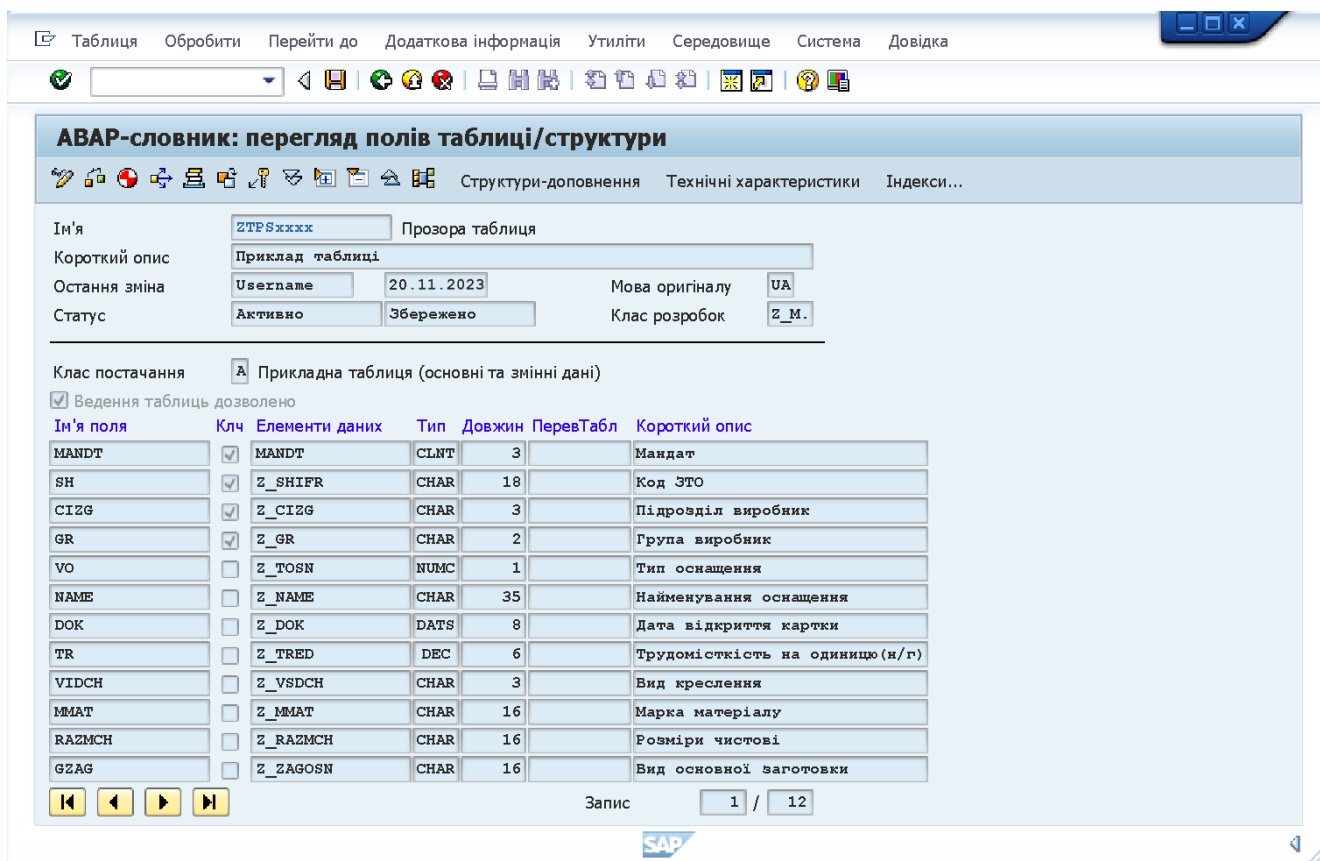


Рисунок 3.2 – АВАР-словник – перегляд полів таблиці

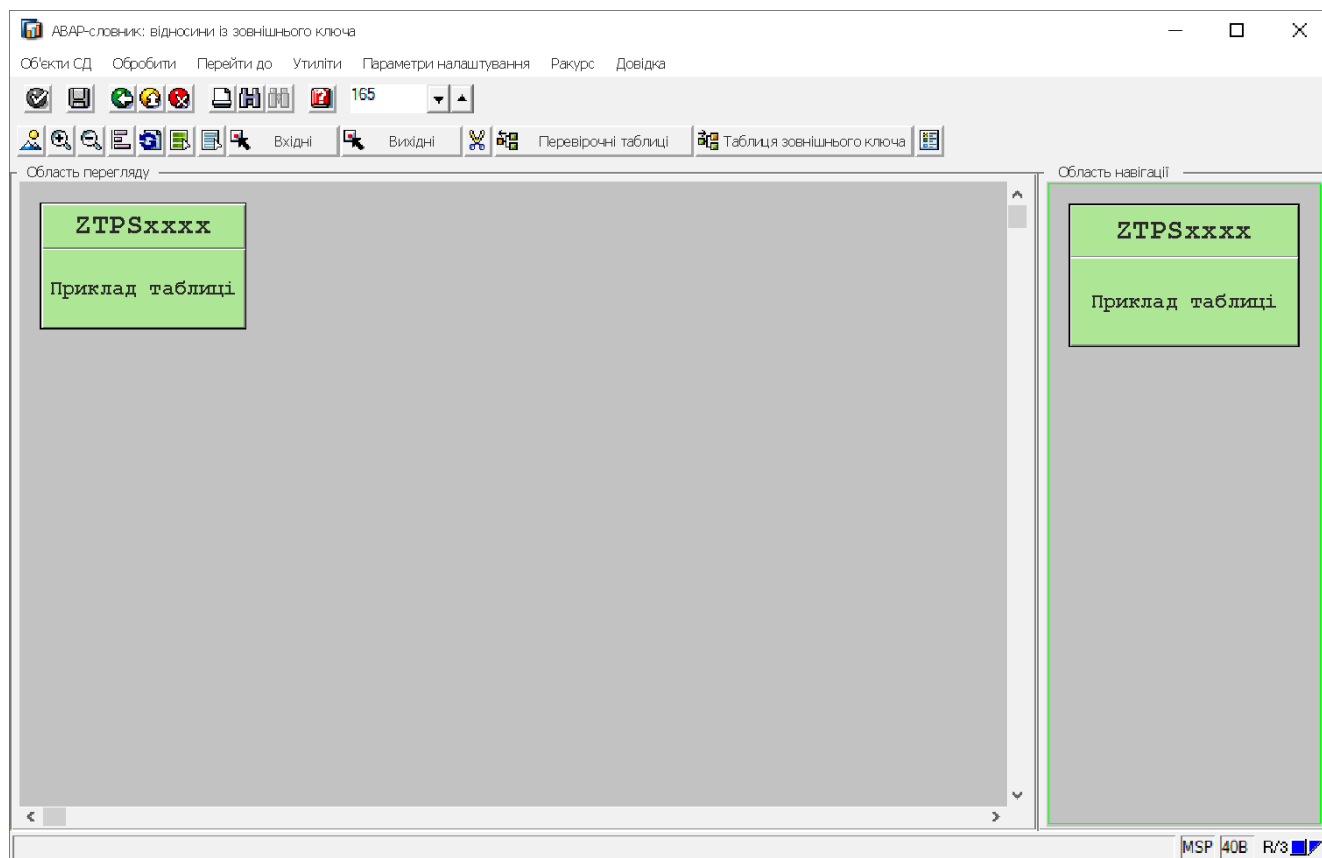


Рисунок 3.3 – АВАР-словник – вікно редактору

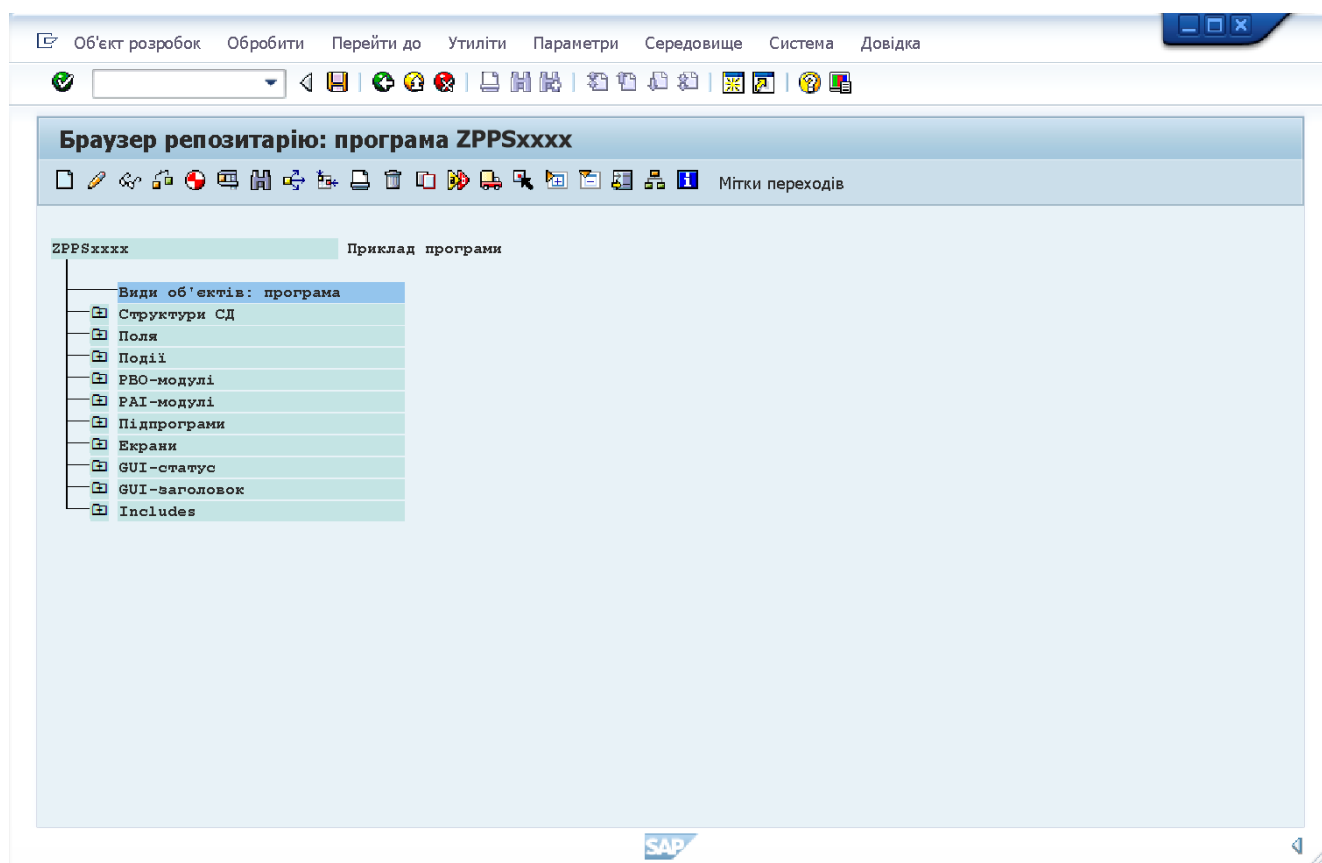


Рисунок 3.4 – Браузер репозитарію

3.1.2 Структурна схема бази даних

Структурна схема бази даних та взаємодія таблиць показана на рис. 3.5.

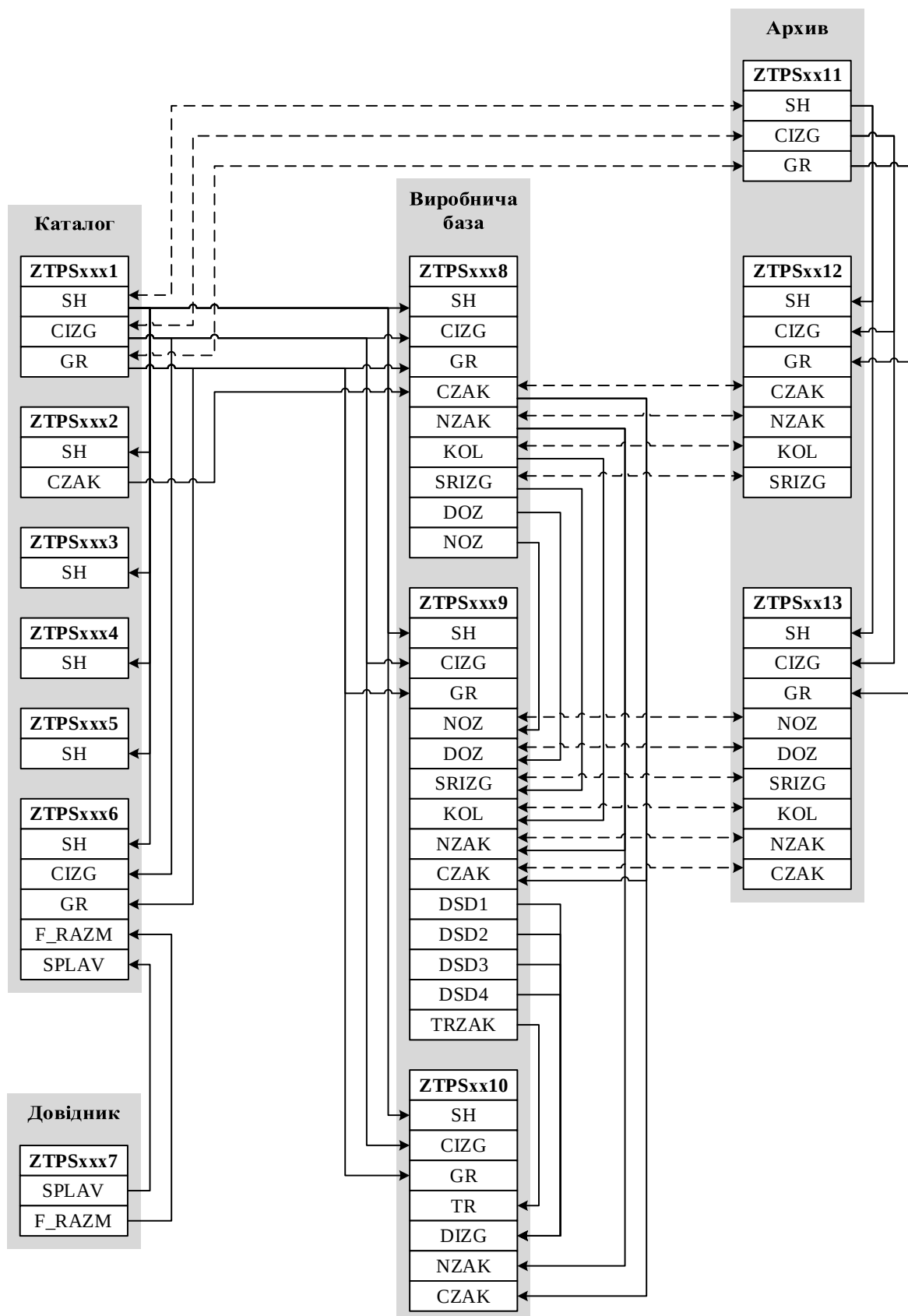


Рисунок 3.5 – Структурна схема бази даних та взаємодія таблиць

3.1.3 Опис необхідних таблиць

Основна таблиця картотеки інструменту та оснащення (каталог) – ZTPSxxx1. Містить в собі дані по коду та найменуванню ЗТО, підрозділу, який виготовляє данні ЗТО, типу ЗТО. Ключові поля у таблиці – SH, CIZG, GR.

Таблиця підрозділів замовників для яких виготовляється ЗТО – ZTPSxxx2. Містить інформацію щодо виробів для яких використовуються данні ЗТО. Ключові поля у таблиці – SH, CZAK NZAK, NDET, OPERATION.

Таблиця додаткового інструменту та оснащення, які входять до складу основного ЗТО – ZTPSxxx3. Ключові поля у таблиці – SH.

Таблиця норм витрат інструменту, який знаходиться на лімітній системі забезпечення – ZTPSxxx4. Ключові поля у таблиці – SH, CPOT.

Таблиця додаткового інструменту та оснащення, які працюють у комплекті з основним ЗТО – ZTPSxxx5. Ключові поля у таблиці – SH.

Таблиця допоміжних матеріалів, необхідних для виготовлення ЗТО – ZTPSxxx6. Ключові поля у таблиці – SH, CIZG, GR, ZAGVSP, PODR2, F_RAZM, SPLAV.

Довідник – ZTPSxxx7. Містить довідникову інформацію щодо матеріалів, які використовуються для виробництва ЗТО. Ключові поля у таблиці – SPLAV, F_RAZM.

Таблиця заявок від підрозділів замовників на виготовлення ЗТО – ZTPSxxx8. Містить дані про дату отримання заявки на виготовлення, кількість та дату, до якої необхідно виготовити ЗТО. Ключові поля у таблиці – SH, CIZG, GR, NZ.

Таблиця замовлень на виготовлення ЗТО, які знаходяться у виробництві ЗТО – ZTPSxxx9. Містить інформацію щодо стану замовлення у виробництві (етапи виробництва). Ключові поля у таблиці – SH, CIZG, GR, NOZ.

Таблиця виконаних замовлень – ZTPSxx10. Містить інформацію щодо виготовлених ЗТО згідно замовлень. Ключові поля у таблиці – SH, CIZG, GR, NPOZ.

Архів записів, видалених з картотеки – ZTPSxx11. Ключові поля у таблиці – SH, CIZG, GR.

Архів записів, видалених з бази заявок – ZTPSxx12. Ключові поля у таблиці – SH, CIZG, GR, CZAK, TIPZAP, VZ, NZAK, NMER, SRIZG, DPZ, KOL.

Архів записів, видалених з бази замовлень – ZTPSxx13. Ключові поля у таблиці – SH, CIZG, GR, NOZ.

3.2 Розробка програмного забезпечення

Після створення всіх необхідних таблиць, починаємо писати код програми у АВАР-редакторі (рис. 3.6).

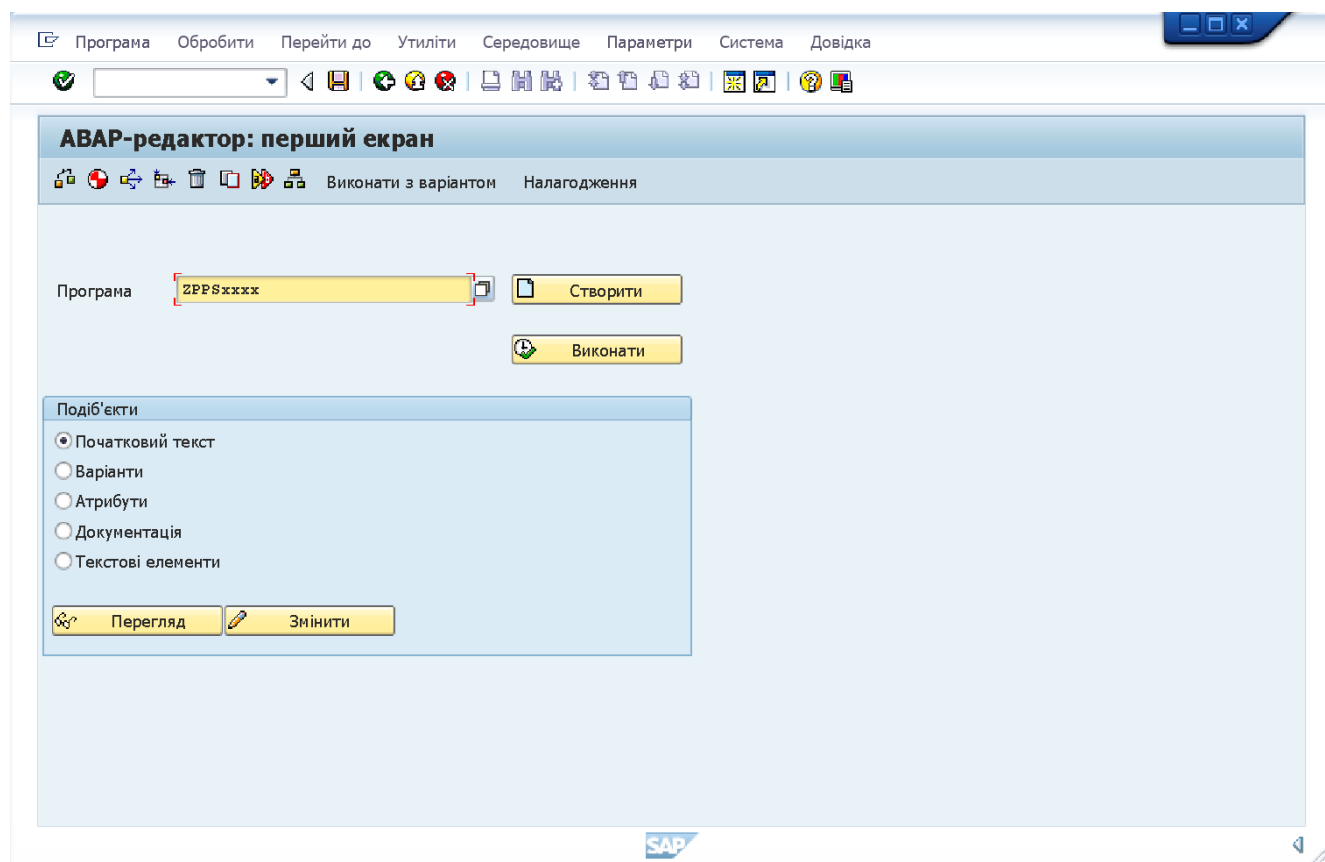


Рисунок 3.6 – Початковий екран редактору програм

Для внесення даних у каталог ЗТО розробляємо програму створення картки ЗТО – ZPPSxxx1. Частковий лістинг програми наведено у Додатку А.

Розробляємо програму ведення виробничої бази – ZPPSxxx2. Частковий лістинг програми наведено у Додатку Б.

Для розрахунку плану виробництва ЗТО розробляємо програму формування плану виробництва – ZPPSxxx3. Частковий лістинг програми наведено у Додатку В.

Для формування плану виробництва до кожного коду ЗТО потрібно врахувати наступні параметри:

- трудомісткість виготовлення ЗТО;
- місяць, у якому потрібно виготовити ЗТО;
- пріоритет виготовлення ЗТО.

Для кожного коду ЗТО розраховуємо термін виготовлення по формулі 3.1.

При обчисленні терміну виготовлення для кожного ЗТО необхідно врахувати, що місячна трудомісткість виготовлення одного коду ЗТО не повинна перевищувати 300 нормо-годин.

$$n_{\text{ВИГ.}} = \frac{T_{\text{ЗТО}}}{T_{\text{М}}}, \quad (3.1)$$

де $n_{\text{ВИГ.}}$ – термін виготовлення, місяць;

$T_{\text{ЗТО}}$ – трудомісткість виготовлення, нормо-година;

$T_{\text{М}}$ – місячна трудомісткість виготовлення одного коду ЗТО, нормо-година ($T_{\text{М}} = 300$).

Виходячи з терміну виготовлення, розраховуємо місяць, в якому потрібно включити код ЗТО до плану виробництва, по формулі 3.2.

$$M_{\text{В.П.}} = M_{\text{ВИГ.}} - n_{\text{ВИГ.}}, \quad (3.2)$$

де $M_{\text{В.П.}}$ – місяць, в якому потрібно включити код ЗТО до плану виробництва;

$M_{\text{виг.}}$ – місяць, у якому потрібно виготовити ЗТО.

Трудомісткість виготовлення усіх кодів ЗТО, які будуть включені до плану виробництва ($T_{\text{заг.}}$), не повинна перевищувати потужність підрозділу, який виготовляє ЗТО (T), тобто $T_{\text{заг.}} \leq T$, у ідеальному випадку $T_{\text{заг.}} = T$.

Пріоритет виготовлення ЗТО (P) поділяється на дві черги: перша черга на виготовлення ($P = 1$), та друга черга ($P = 2$).

План виробництва ЗТО формується на відповідний місяць (M).

Блок-схема алгоритму формування плану виробництва ЗТО показана на рис. 3.7.

Для усіх програм реалізований режим, розрахований на одночасну роботу великої кількості користувачів. Для цього виконується блокування окремого запису каталогу бази даних.

При наповненні інформацією таблиць бази даних, використовуємо чотири методи заповнення полів:

- ручне внесення даних – спеціально навчений спеціаліст за допомогою інтерфейсу заповнює необхідні поля;
- ручний вибір даних з drop-down list;
- автоматичне заповнення полів інформацію з таблиць пов'язаних ключовими полями;
- автоматичне заповнення поточної дати та ID користувача.

Тестування розроблених програм буде виконуватись користувачами у навчальному середовищі системи. Для користувачів створюється окремий «клієнт» для тестування. Налагодження цього «клієнту» виконується у середовищі 210: SAND [6]. Доступ до тестування програм прописаний у налаштуванні клієнта.

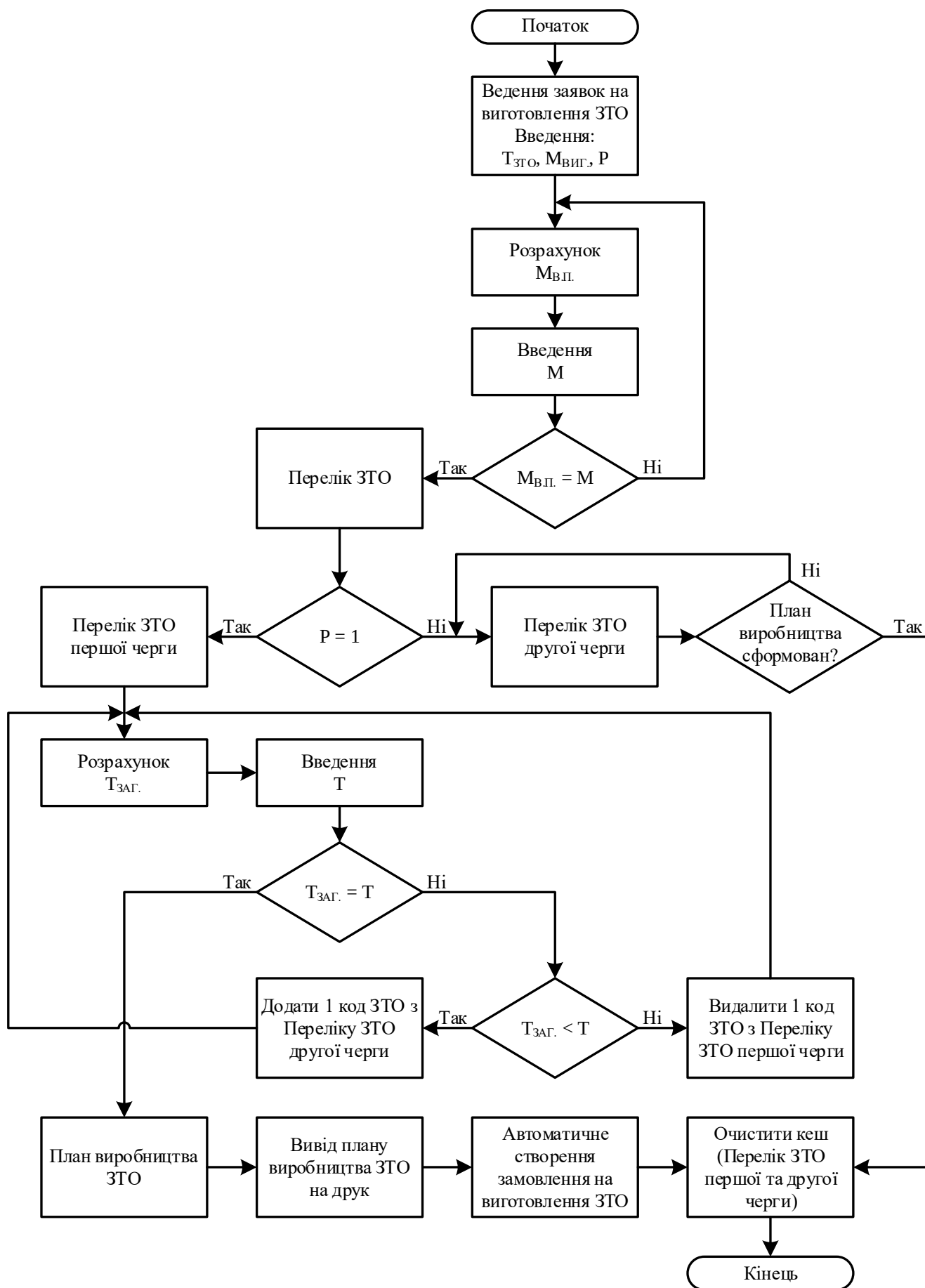


Рисунок 3.7 – Блок-схема алгоритму формування плану виробництва ЗТО

За необхідністю підтримки існуючого каталогу ЗТО та виробничої бази у Clipper, вся внесена у розроблюваних програмах інформація копіюється у *.dbf-файли. Автоматичне завантаження інформації відбувається пакетним процесом у відповідний час по графіку запрограмованих задач.

3.3 Розробка інтерфейсів користувача

Редактор екранів дозволяє визначати дизайн екранів користувача, а також послідовність динамічних програм. Екран інтерпретується не тільки АВАР/4, але й окремим інтерпретатором динамічних програм DYNPRO – повноцінним середовищем для швидкої розробки прототипів та завдання послідовності їх виконання.

Редактор екранів визначає наступні характеристики:

- програмні атрибути екрану, у тому числі номер екрану, тип екрану (нормальний, діалоговий, додатковий) та ін.;
- зовнішній вид екрану, у тому числі розташування полів, ярлики, кнопки прокрутки, значки, галочки та групові елементи;
- поля баз даних, перехресні підтвердження та ін.;
- логічну послідовність.

Для створення дизайну екрану у редакторі передбачені графічні та букво-цифрові варіанти редактору. У графічному варіанті редактору ярлики полів, їх положення та інші характеристики можна змінювати прямо на екрані, при цьому увесь текст екрану може бути представлений графічними позначками. У букво-цифровому варіанті редактору дизайн можна реалізувати тільки за допомогою можливостей меню редактору [8].

Розробка екранів користувача у даному проекті буде виконуватись букво-цифровим варіантом.

Приклад роботи у редакторі екрану наведений на рис. 3.8-3.10.

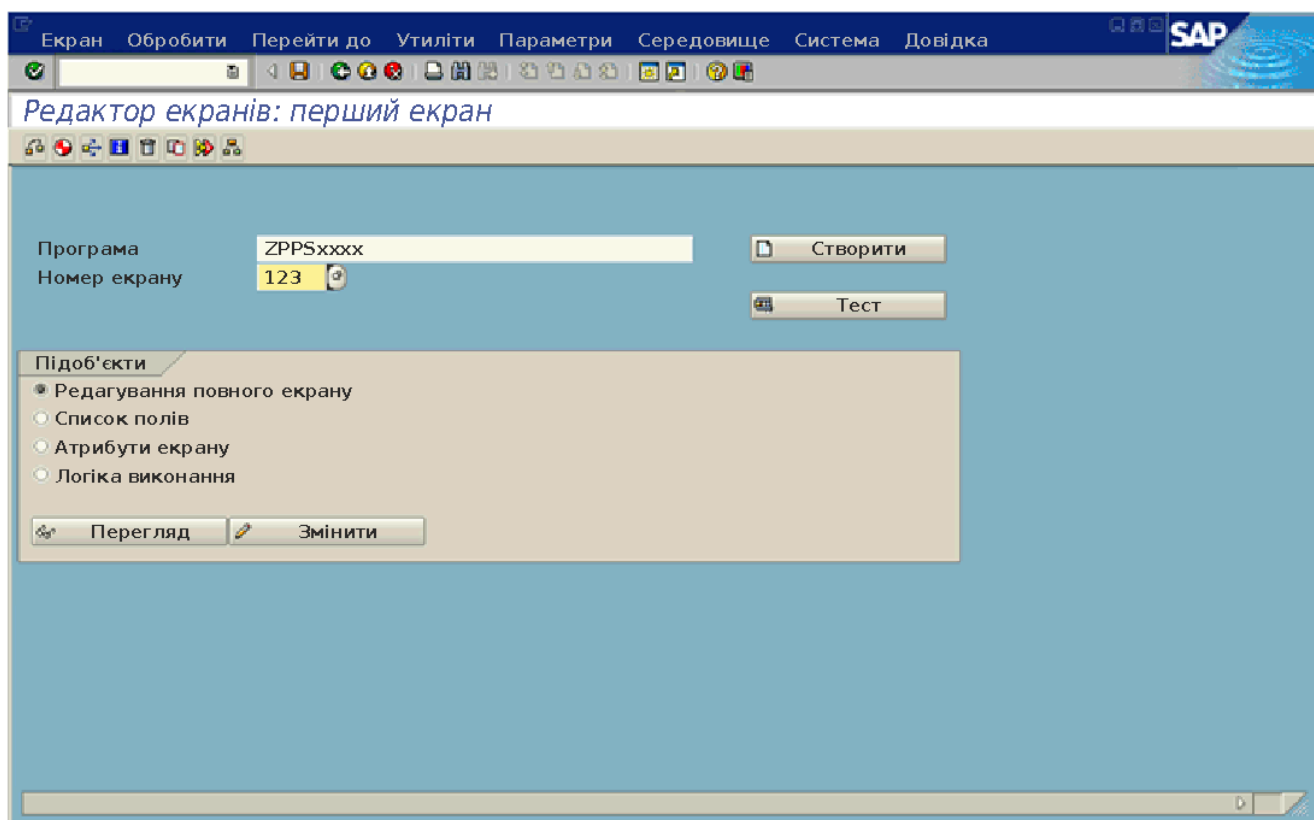


Рисунок 3.8 – Перший екран редактора екранів

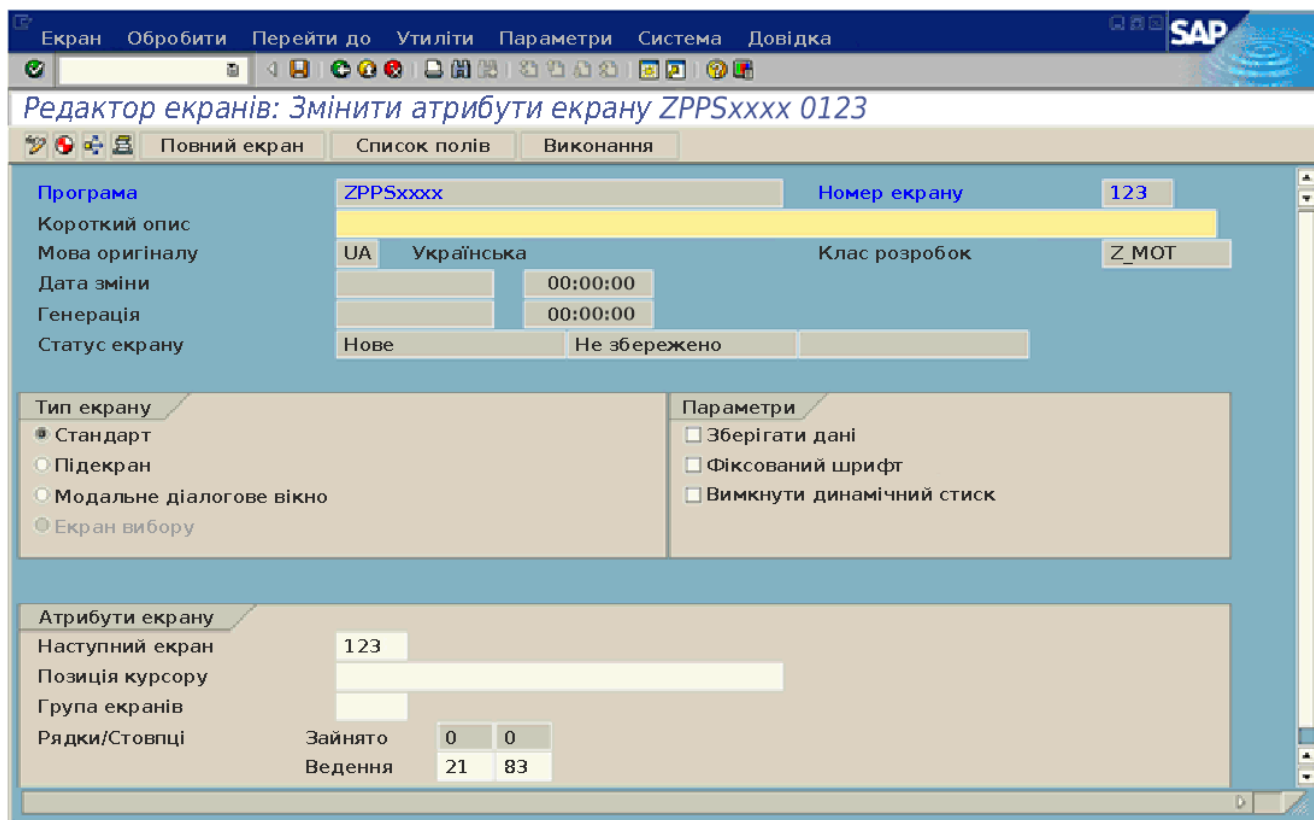


Рисунок 3.9 – Зміна атрибутів у редакторі екранів

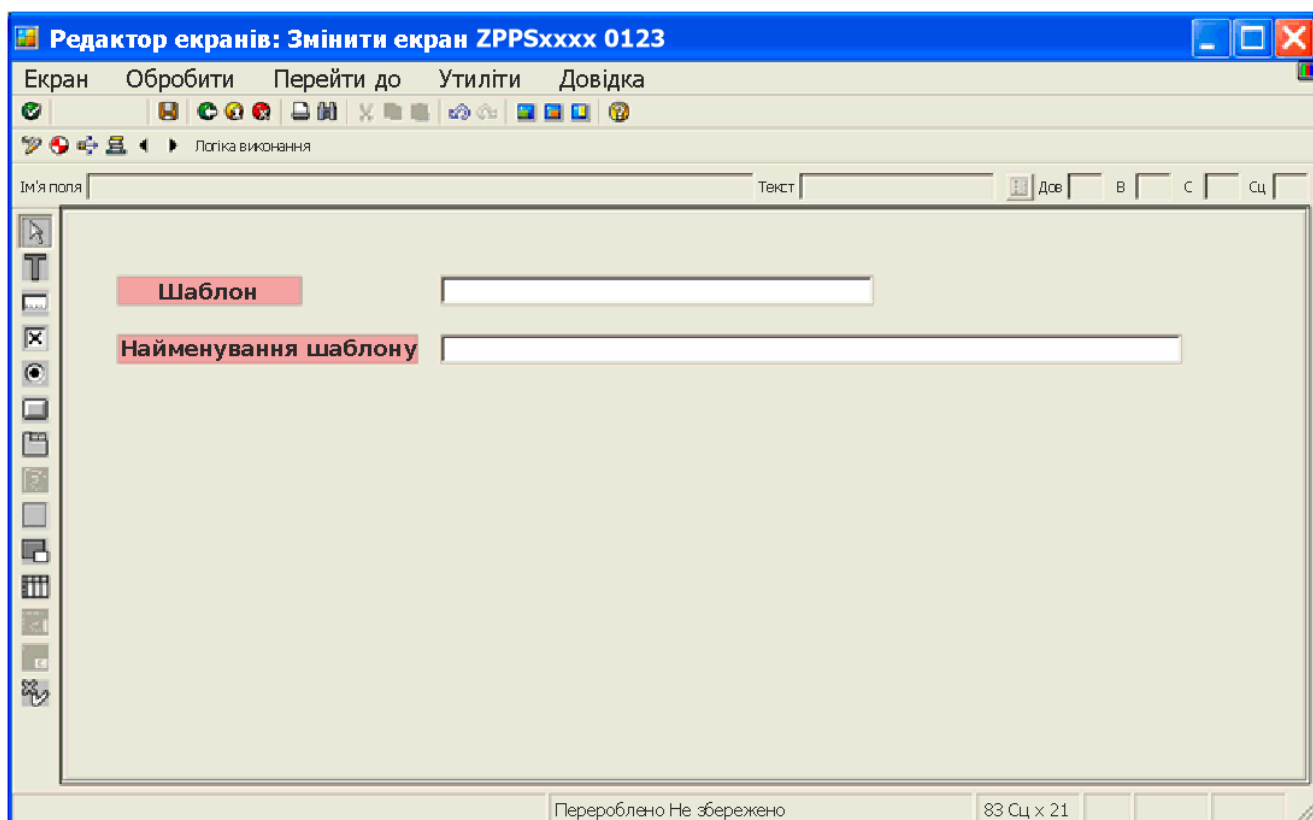


Рисунок 3.10 – Створення екрану у редакторі екранів

3.4 Опис програм

3.4.1 Програма створення картки ЗТО

Програма ZPPSxxx1 призначена для створення картки коду ЗТО. Створена картка зберігається у каталог системи SAP, який виступає у ролі єдиної бази ЗТО на підприємстві. Програма дозволяє створити новий запис у каталозі використовуючи певний шаблон формату коду, який регламентується відповідною інструкцією (нормативною документацією). Об'єм інформації, необхідної для створення картки – мінімальний. Інтерфейс користувача програми складається з одного робочого вікна. Всі поля є обов'язковими для заповнення. Поля «Код» та «Найменування» вносяться методом ручного вводу даних. Дата відкриття картки заповнюється автоматично поточною датою, і разом з іменем користувача прив'язується до каталогу ЗТО. Інші поля заповнюються методом

вибору даних із drop-down list. Зовнішній вигляд інтерфейсу користувача представлений на рис. 3.11.

У допоміжному виробництві один і той же код ЗТО можуть виготовляти різні виробничі групи різних підрозділів, тому у програмі реалізовано створення однакової картки ЗТО для різних виробників. Подальший вибір виробника при внесенні заявки на виготовлення залежить від завантаженості відповідної виробничої групи.

Рисунок 3.11 – Вікно «Створення картки у каталозі»

3.4.2 Програма ведення виробничої бази

Програма ZPPSxxx2 призначена для ведення заявок та замовлень на виготовлення ЗТО. Для створення заявки на виготовлення, користувач програми повинен знайти необхідний код ЗТО у каталозі, обрати запис за необхідним підрозділом та групою виробником. Зовнішній вигляд інтерфейсу користувача представлений на рис. 3.12-3.19.

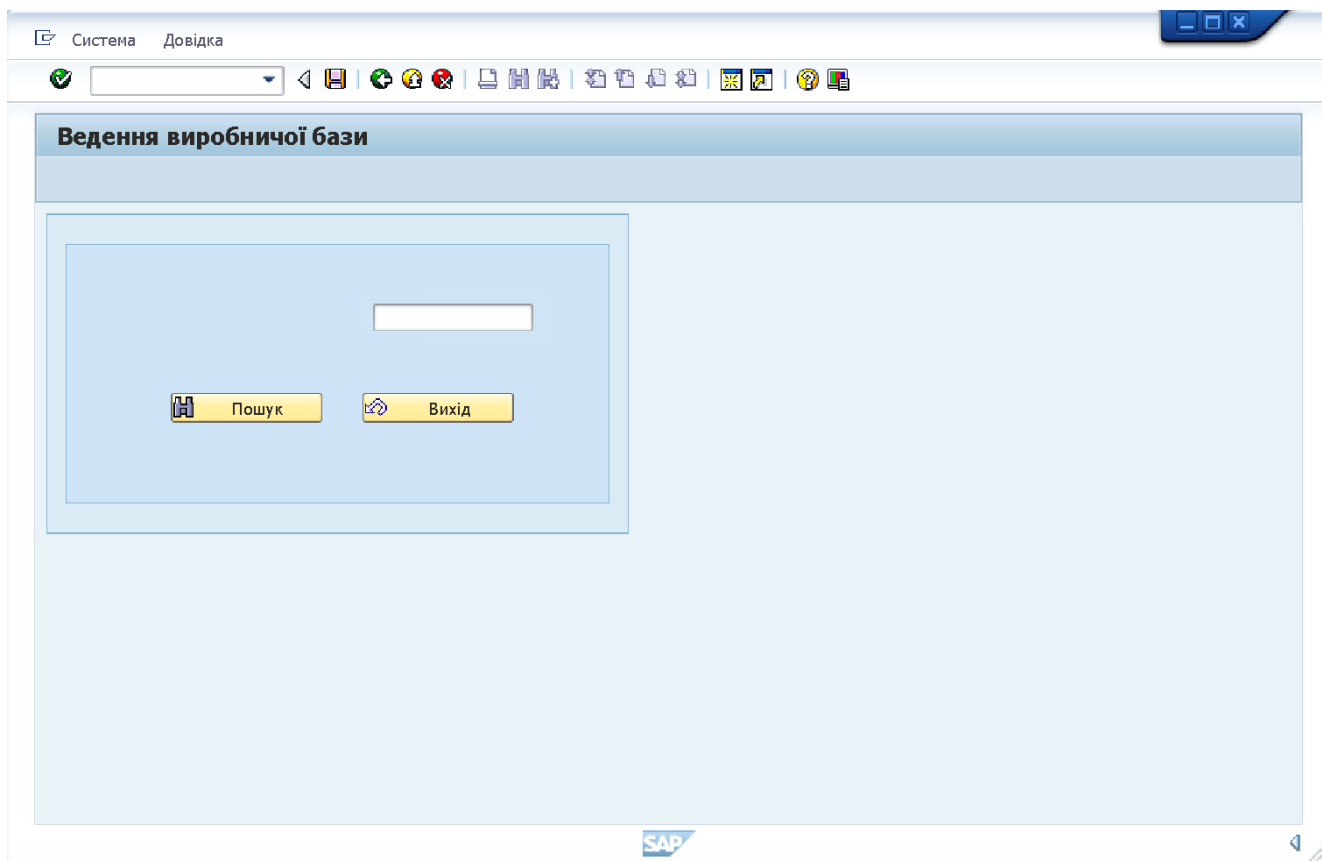


Рисунок 3.12 – Початковий екран

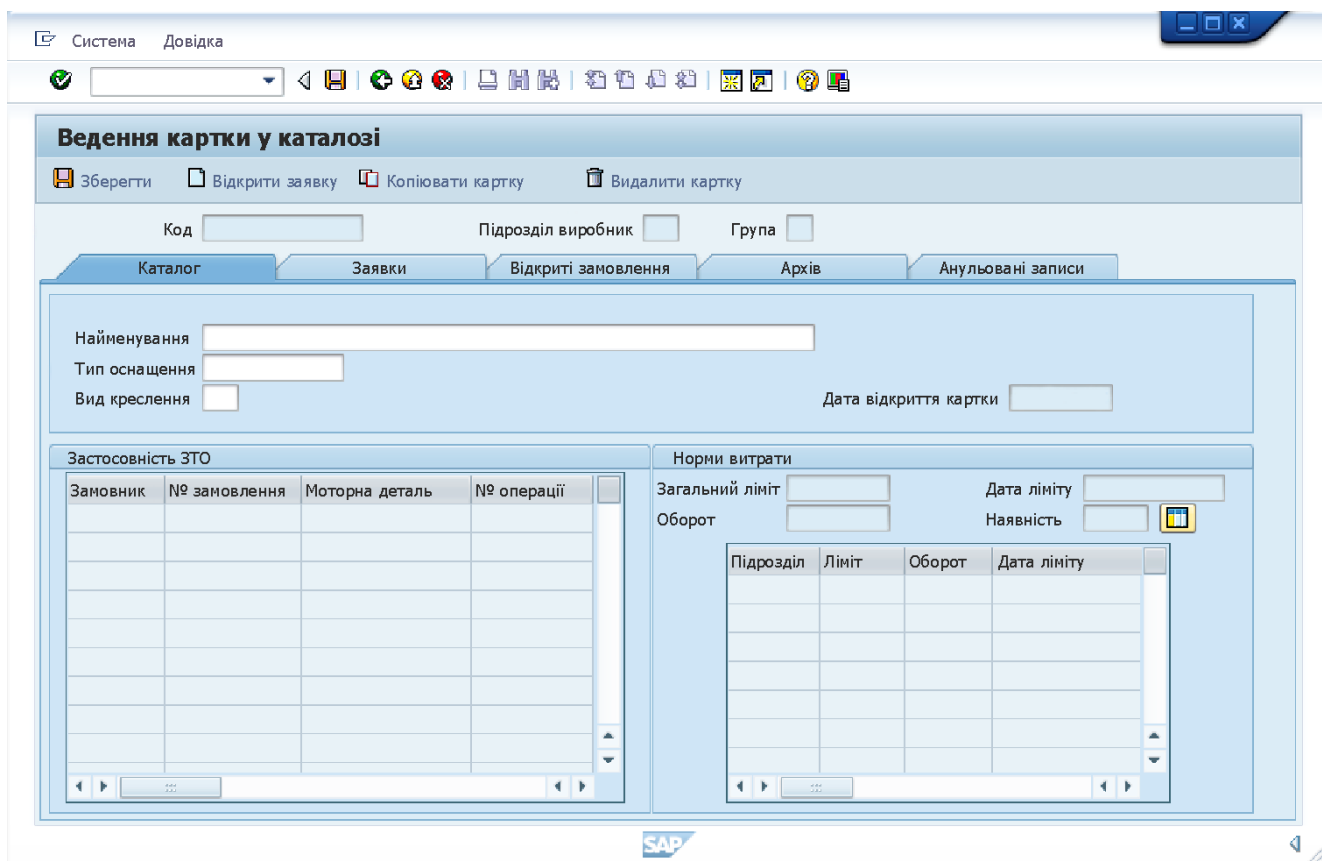


Рисунок 3.13 – Вкладка «Каталог»

[illegible]

Рисунок 3.15 – Вкладка «Заявки»

Замовлення на виготовлення	
Код	Підрозділ виробник Група
Найменування	
Вид замовлення	Підрозділ замовник
Тип документу	Дата відкриття замовлення
Термін виготовлення	Кількість Од.вим.
Номер замовлення	Трудомісткість
Пріоритет виготовлення	

Стан у виробництві	Відомості по здачі
Дата надходження до технічного відділу	1. Дата здачі 1. Кількість здачі
Дата надходження до ВДБ	2. Дата здачі 2. Кількість здачі
Дата надходження до заготівельного підрозділу	3. Дата здачі 3. Кількість здачі
Дата надходження до філії	4. Дата здачі 4. Кількість здачі
Дата надходження до групи	

Рисунок 3.17 – Вікно «Замовлення на виготовлення»

[illegible]

Рисунок 3.19 – Вкладка «Анульовані записи»

Інтерфейс користувача програми ведення виробничої бази складається з восьми робочих вікон, п'ять з яких реалізовані у форматі вкладок.

У програмі реалізовані наступні функції:

- простий у використанні пошук за кодом/ початком коду ЗТО;
- перегляд всіх кодів ЗТО, які доступні у каталозі;
- ведення деталізованої інформації про ЗТО – назва, матеріал та інші технічні характеристики;
- можливість перегляду залишків ЗТО на складі через модуль ММ;
- ведення додаткової бази підрозділів, які використовують конкретні ЗТО для конкретних виробів.

Відкриття замовлення на виготовлення ЗТО виконується автоматично на основі існуючої заявки від підрозділу замовника. Також є можливість перегляду відповідної заявки на виготовлення ЗТО та відкритого замовлення з усією додатковою інформацією. Після виробництва ЗТО за відкритим замовленням, запис переноситься до вкладки «Архів». Всі записи відносно заявок та замовлень, які втратили свою актуальність можливо перемістити у вкладку «анульовані записи» з можливістю їх подальшого відновлення.

3.4.3 Програма формування плану виробництва

Програма призначена для формування плану виробництва для конкретної виробничої групи конкретного підрозділу виробника. Інтерфейс користувача програми формування плану виробництва складається з одного робочого вікна, зовнішній вигляд якого представлений на рис. 3.20. Для формування плану користувач повинен вказати підрозділ та виробничу групу, для якої буде формуватися план. Також треба обрати місяць, на який формується план виробництва. У допоміжному виробництві план виробництва формується у поточному місяці на наступний, але при необхідності є можливість формувати план на будь-який місяць року. Дані за полем «Потужність групи у нормо-годинах» заповнюється автоматично з бази модулю ММ. Потужність залежить від чисельності виробничої групи та кількості робочих днів у місяці. Потужність групи у кодах ЗТО вказується користувачем вручну і є додатковим критерієм для

вибору кодів для включення в план виробництва. Два останні критерії помічаються при необхідності термінового виготовлення ЗТО або поповнення запасів складу оснащенням. Після введення цих даних, формується перелік кодів ЗТО для попереднього перегляду (рис. 3.21). Варто зазначити, що перелік кодів ЗТО для включення у план виробництва формується не лише з наявних заявок на виготовлення, але й містить вже відкриті замовлення. При експортуванні переліку у Excel, програма автоматично відкриває замовлення на існуючі заявки, дата вже відкритих замовлень не змінюється. Також у виробничу базу записується код програми виробництва, для відстеження виготовлення ЗТО.

The screenshot shows a SAP window titled 'Система Довідка' (System Reference). The main content area is titled 'Формування плану виробництва' (Formulation of production plan). It contains a form with the following fields and options:

- Підрозділ виробник** (Sub-division manufacturer): A text input field.
- Група** (Group): A text input field.
- Формування плану на** (Formulation of plan for): A text input field followed by a dropdown menu set to 'місяць' (month).
- Потужність групи у нормо-годинах** (Power of group in normo-hours): A numeric input field.
- Потужність групи у кодах ЗТО** (Power of group in ZTO codes): A numeric input field.
- Включати у план коди ЗТО** (Include ZTO codes in plan): Two checkboxes:
 - ☐ не забезпечені матеріалом (not provided with material)
 - ☐ для забезпечення згідно норм витрат (for provision according to consumption norms)
- Сформувати план виробництва** (Formulate production plan): A button with a small icon to the left.

The SAP logo is visible in the bottom right corner of the window.

Рисунок 3.20 – Вікно «Формування плану виробництва»

Система Довідка

Створення картки у каталозі

Зберегти

Код

Найменування

Тип оснащення

Вид креслення Дата відкриття картки

Підрозділ виробник Група

SAP

Рисунок 3.22 – Створення картки у каталозі

Система Довідка

Ведення виробничої бази

Пошук коду ЗТО

Код	Підрозділ виробник	Група
УХХХХ-0001	В01	Г1
УХХХХ-0001-01	В01	Г1
УХХХХ-0001-02	В01	Г1
УХХХХ-0002	В03	Г2
УХХХХ-0003	В01	Г1
УХХХХ-0004	В02	Г5
УХХХХ-0005	В02	Г3
УХХХХ-0006	В04	Г2
УХХХХ-0006-01	В04	Г2
УХХХХ-0007	В03	Г3
УХХХХ-0008	В02	Г1
УХХХХ-0009	В02	Г2

Пошук Вихід

Вибрати Скасувати

SAP

Рисунок 3.23 – Пошук необхідної картки ЗТО

Ведення картки у каталозі

Зберегти Відкрити заявку Копіювати картку Видалити картку

Код: YXXXX-0001 Підрозділ виробник: B01 Група: Г1

Каталог Заявки Відкриті замовлення Архів Анульовані записи

Найменування: ІНСТРУМЕНТ СПЕЦІАЛЬНИЙ
 Тип оснащення: Тип №1
 Вид креслення: ТВЧ
 Дата відкриття картки: 01.01.2023

Замовник	№ замовлення	Моторна деталь	№ операції
З01	хх.001	MD-001.XXXXX	005
З02	хх.002	MD-002.XXXXX	010
З02	хх.003	MD-003.XXXXX	015
З02	хх.004	MD-004.XXXXX	020
З03	хх.003	MD-003.XXXXX	025
З04	хх.004	MD-004.XXXXX	030

Підрозділ	Ліміт	Оборот	Дата ліміту
З01	5	15	01.08.2023
З02	20	60	01.06.2023
З03	10	25	01.09.2023
З04	1.2	5	18.01.2023

Рисунок 3.24 – Первинна інформація по коду ЗТО

Для створення заявки на виготовлення ЗТО, у вкладці «Каталог» за допомогою кнопки «Відкрити заявку», відкриваємо вікно «Заявка на виготовлення». У вікні заповнюємо усі необхідні поля. Вказаний термін виготовлення впливає на подальше включення коду ЗТО у план виробництва. Приклад заповнення наведений на рис. 3.25.

Для перегляду існуючих заявок на заданий код ЗТО використовуємо вкладку «Заявки» у якій, шляхом виділення необхідного запису, можливо змінити введені раніше дані або видалити запис. Приклад переліку існуючих заявок наведений на рис. 3.26.

Рисунок 3.25 – Заповнена заявка на виготовлення ЗТО

[illegible]

Рисунок 3.26 – Перелік існуючих заявок на виготовлення

Після включення заявки на виготовлення в план виробництва автоматично формується замовлення на виготовлення ЗТО. Для перегляду відкритих замовлень на заданий код ЗТО використовуємо вкладку «Відкриті замовлення». Приклад переліку відкритих замовлень наведений на рис. 3.27. У цій вкладці, шляхом виділення необхідного запису, можливо переглянути детальний стан замовлення у виробництві або видалити запис (рис. 3.28).

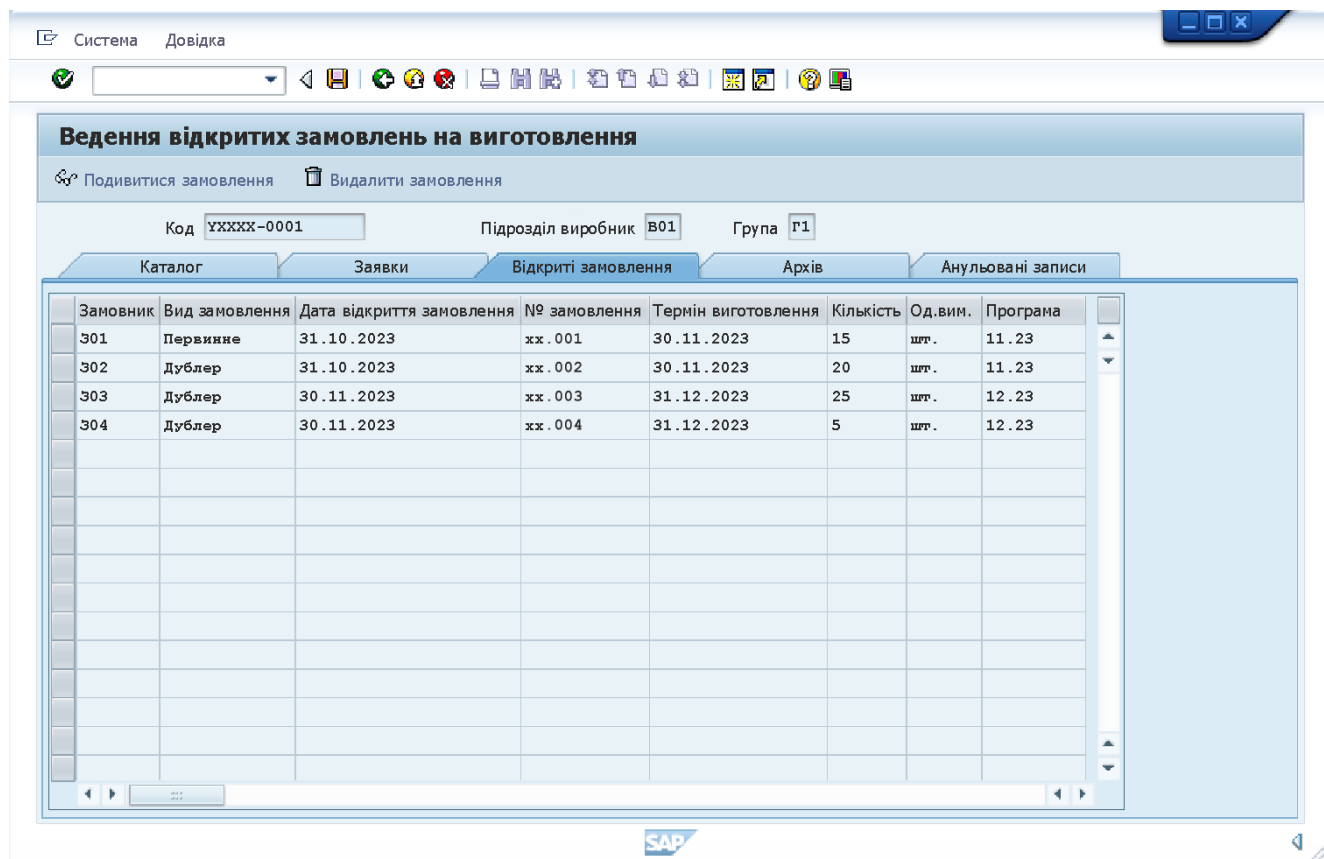


Рисунок 3.27 – Перегляд відкритих замовлень на виготовлення

Після виготовлення коду ЗТО замовлення на виготовлення автоматично закривається, відповідний запис переноситься до архіву. Переглянути всі виконані замовлення по коду ЗТО можливо у вкладці «Архів». Приклад переліку виконаних замовлень наведений на рис. 3.29.

Видалені записи заявок та замовлень можливо переглянути у вкладці «Анульовані записи». За потреби видалені записи можливо відновити. Приклад переліку видалених записів наведений на рис. 3.20.

Рисунок 3.28 – Перегляд стану замовлення у виробництві

[illegible]

Рисунок 3.29 – Перегляд архіву виконаних замовлень

Замовник	Вид замовлення	№ замовлення	Термін виготовлення	Кількість	Од.вим.	Трудомісткість	Дата анулювання	Причина анулювання
305	Дублер	хх.005	31.10.2022	10	шт.	30.0	12.10.2022	на вимогу замовника
306	Первинне	хх.006	30.12.2022	15	шт.	45.0	09.11.2022	втрата актуальності
307	Дублер	хх.007	30.12.2022	15	шт.	45.0	23.11.2022	анулювання виробу
308	Дублер	хх.008	27.01.2023	5	шт.	15.0	15.12.2022	підбір
309	Ремонт	хх.009	24.02.2023	8	шт.	24.0	17.01.2023	розпорядження

Рисунок 3.30 – Перегляд анульованих записів

Слід зазначити що, при видаленні записів необхідно обрати одну з наступних причин видалення даних: на вимогу замовника, втрата актуальності, анулювання виробу, підбір, розпорядження, помилкова інформація. Дані, видалені за причиною «Помилкова інформація» після видалення не розміщуються в архіві видалених записів, та не можуть бути відновлені.

Для створення плану виробництва у робочому вікні програми ZPPSxxx3 заповнюємо усі необхідні поля. Приклад заповнення наведений на рис. 3.31. Натискаємо кнопку «Сформувати план виробництва» та отримуємо вікно попереднього перегляду сформованого плану (рис. 3.32). Кнопка «Експорт у Excel» автоматично відкриває додаток MS Office для подальшого збереження сформованого плану. Приклад наведений на рис. 3.33.

Система Довідка

Формування плану виробництва

Підрозділ виробник: B01 Група: Г1

Формування плану на: 11 місяць

Потужність групи у нормо-годинах: 1500.0

Потужність групи у кодах ЗТО: 35

Включати у план коди ЗТО

☐ не забезпечені матеріалом

☐ для забезпечення згідно норм витрат

Сформувати план виробництва

SAP

Рисунок 3.31 – Формування плану виробництва за заданими параметрами

Система Довідка

План виробництва

Підрозділ виробник: B01 Група: Г1 План на 11 місяць Експорт у Excel

№ з/п	Код ЗТО	Підрозділ замовник	№ замовлення	Кількість	Трудо-місткість	Стан			Термін виготовлення	Примітка
						техвідділ	п.забезп.	група		
1	УХХХХ-0001	301	хх.001	15	45.0	20.09.2023	30.09.2023	15.10.2023	30.11.2023	
2	УХХХХ-0001	302	хх.002	20	60.0	20.09.2023	30.09.2023	15.10.2023	30.11.2023	
3	УХХХХ-0002	302	хх.002	3	300.0	02.10.2023	12.10.2023	27.10.2023	15.11.2023	
4	УХХХХ-0003	303	хх.003	20	30.0	30.09.2023	10.10.2023	25.10.2023	30.11.2023	
5	УХХХХ-0004	304	хх.004	5	20.0	15.09.2023	25.09.2023	10.10.2023	30.11.2023	
6	УХХХХ-0005	305	хх.005	8	50.0	23.09.2023	03.10.2023	18.10.2023	30.11.2023	
7	УХХХХ-0006	306	хх.006	2	24.0	30.09.2023	10.10.2023	25.10.2023	30.11.2023	
8	УХХХХ-0007	307	хх.007	50	80.0	30.09.2023	10.10.2023	25.10.2023	30.11.2023	
9	УХХХХ-0008	308	хх.008	24	40.0	29.09.2023	09.10.2023	24.10.2023	30.11.2023	
10	УХХХХ-0009	309	хх.009	1	15.0	02.10.2023	12.10.2023	27.10.2023	30.11.2023	
11	УХХХХ-0010	310	хх.010	1	15.0	05.10.2023	15.10.2023	30.10.2023	30.11.2023	
12	УХХХХ-0011	311	хх.011	4	30.0	05.10.2023	15.10.2023	30.10.2023	30.11.2023	
13	УХХХХ-0012	312	хх.012	7	7.0	24.09.2023	04.10.2023	19.10.2023	30.11.2023	
14	УХХХХ-0013	313	хх.013	15	20.0	01.10.2023	11.10.2023	26.10.2023	30.11.2023	
15	УХХХХ-0014	314	хх.014	30	15.0	18.09.2023	28.09.2023	13.10.2023	30.11.2023	
16	УХХХХ-0015	315	хх.015	6	3.0	17.09.2023	27.09.2023	12.10.2023	30.11.2023	
17	УХХХХ-0016	316	хх.016	2	50.0	25.09.2023	05.10.2023	20.10.2023	30.11.2023	
18	УХХХХ-0017	317	хх.017	10	20.0	25.09.2023	05.10.2023	20.10.2023	30.11.2023	
19	УХХХХ-0018	318	хх.018	8	16.0	16.09.2023	26.09.2023	11.10.2023	30.11.2023	
20	УХХХХ-0019	319	хх.019	2	5.0	30.09.2023	10.10.2023	25.10.2023	30.11.2023	
21	УХХХХ-0020	320	хх.020	5	25.0	29.09.2023	09.10.2023	24.10.2023	30.11.2023	
22	УХХХХ-0021	321	хх.021	7	7.0	21.09.2023	01.10.2023	16.10.2023	30.11.2023	
23	УХХХХ-0021	322	хх.022	2	4.0	21.09.2023	01.10.2023	16.10.2023	30.11.2023	
24	УХХХХ-0022	322	хх.022	4	4.0	17.09.2023	27.09.2023	12.10.2023	30.11.2023	
25	УХХХХ-0023	323	хх.023	10	10.0	02.09.2023	12.09.2023	27.09.2023	30.11.2023	
26	УХХХХ-0024	324	хх.024	25	20.0	11.09.2023	21.09.2023	06.10.2023	30.11.2023	

SAP

Рисунок 3.32 – Попередній перегляд сформованого плану виробництва

	A	B	C	D	E	F	G	H	I	J	K
1									ЗАТВЕРДЖУЮ:		
2									Керівник виробництва		
3											
4									Ім'я ПРИЗВИЩЕ		
5											
6											
7											
8											
9	№ з/п	Код ЗТО	Підрозділ замовник	№ замовлення	Кількість	Трудо-місткість	Стан у виробництві			Термін виготовлення	Примітка
10							техвідділ	п.забезп.	група		
11	1	YXXXX-0001	301	хх.001	15	45.0	20.09.2023	30.09.2023	15.10.2023	30.11.2023	
12	2	YXXXX-0001	302	хх.002	20	60.0	20.09.2023	30.09.2023	15.10.2023	30.11.2023	
13	3	YXXXX-0002	302	хх.002	3	300.0	02.10.2023	12.10.2023	27.10.2023	15.12.2023	
14	4	YXXXX-0003	303	хх.003	20	30.0	30.09.2023	10.10.2023	25.10.2023	30.11.2023	
15	5	YXXXX-0004	304	хх.004	5	20.0	15.09.2023	25.09.2023	10.10.2023	30.11.2023	
16	6	YXXXX-0005	305	хх.005	8	50.0	23.09.2023	03.10.2023	18.10.2023	30.11.2023	
17	7	YXXXX-0006	306	хх.006	2	24.0	30.09.2023	10.10.2023	25.10.2023	30.11.2023	
18	8	YXXXX-0007	307	хх.007	50	80.0	30.09.2023	10.10.2023	25.10.2023	30.11.2023	
19	9	YXXXX-0008	308	хх.008	24	40.0	29.09.2023	09.10.2023	24.10.2023	30.11.2023	
20	10	YXXXX-0009	309	хх.009	1	15.0	02.10.2023	12.10.2023	27.10.2023	30.11.2023	
21	11	YXXXX-0010	310	хх.010	1	15.0	05.10.2023	15.10.2023	30.10.2023	30.11.2023	
22	12	YXXXX-0011	311	хх.011	4	30.0	05.10.2023	15.10.2023	30.10.2023	30.11.2023	
23	13	YXXXX-0012	312	хх.012	7	7.0	24.09.2023	04.10.2023	19.10.2023	30.11.2023	
24	14	YXXXX-0013	313	хх.013	15	20.0	01.10.2023	11.10.2023	26.10.2023	30.11.2023	
25	15	YXXXX-0014	314	хх.014	30	15.0	18.09.2023	28.09.2023	13.10.2023	30.11.2023	
26	16	YXXXX-0015	315	хх.015	6	3.0	17.09.2023	27.09.2023	12.10.2023	30.11.2023	
27	17	YXXXX-0016	316	хх.016	2	50.0	25.09.2023	05.10.2023	20.10.2023	30.11.2023	
28	18	YXXXX-0017	317	хх.017	10	20.0	25.09.2023	05.10.2023	20.10.2023	30.11.2023	
29	19	YXXXX-0018	318	хх.018	8	16.0	16.09.2023	26.09.2023	11.10.2023	30.11.2023	
30	20	YXXXX-0019	319	хх.019	2	5.0	30.09.2023	10.10.2023	25.10.2023	30.11.2023	
31	21	YXXXX-0020	320	хх.020	5	25.0	29.09.2023	09.10.2023	24.10.2023	30.11.2023	
32	22	YXXXX-0021	321	хх.021	7	7.0	21.09.2023	01.10.2023	16.10.2023	30.11.2023	
33	23	YXXXX-0021	322	хх.022	2	4.0	21.09.2023	01.10.2023	16.10.2023	30.11.2023	
34	24	YXXXX-0022	322	хх.022	4	4.0	17.09.2023	27.09.2023	12.10.2023	30.11.2023	
35	25	YXXXX-0023	323	хх.023	10	10.0	02.09.2023	12.09.2023	27.09.2023	30.11.2023	
36	26	YXXXX-0024	324	хх.024	25	20.0	11.09.2023	21.09.2023	06.10.2023	30.11.2023	
37	27	YXXXX-0025	325	хх.025	1	250.0	05.10.2023	15.10.2023	30.10.2023	30.11.2023	
38	28	YXXXX-0026	326	хх.026	1	100.0	08.09.2023	18.09.2023	03.10.2023	30.11.2023	
39	29	YXXXX-0027	327	хх.027	15	30.0	28.09.2023	08.10.2023	23.10.2023	30.11.2023	
40	30	YXXXX-0028	328	хх.028	7	27.0	25.09.2023	05.10.2023	20.10.2023	30.11.2023	
41	31	YXXXX-0029	329	хх.029	5	50.0	25.09.2023	05.10.2023	20.10.2023	30.11.2023	
42	32	YXXXX-0030	330	хх.030	6	18.0	01.10.2023	11.10.2023	26.10.2023	30.11.2023	
43	33	YXXXX-0031	331	хх.031	10	50.0	05.10.2023	15.10.2023	30.10.2023	30.11.2023	
44	34	YXXXX-0032	332	хх.032	3	60.0	24.09.2023	04.10.2023	19.10.2023	30.11.2023	
45											
46					Загалом:	1500.0					
47											
48	Заступник керівника з планування									Ім'я ПРИЗВИЩЕ	
49											
50	Підготував									Ім'я ПРИЗВИЩЕ	
51											

Готово

Рисунок 3.33 – Сформований план виробництва у додатку MS Office

ВИСНОВКИ

Дипломний проект націлений на модернізацію системи планування виробництва інструменту та оснащення на базі Clipper до більш ефективної і сучасної системи ERP. Основна мета проекту полягала у перенесенні функціоналу з однієї системи до іншої з метою оптимізації процесів виробництва та поліпшення їх інтеграції з існуючими модулями ERP.

У ході виконання роботи проведений ретельний аналіз проблеми і обґрунтована актуальність впровадження нової системи планування. Він чітко визначає недоліки і обмеження існуючої системи, зокрема її застарілість та неефективність у вирішенні сучасних виробничих завдань. Також відзначається важливість швидкості, точності та гнучкості у сучасному управлінні виробництвом.

У проекті описаний процес перенесення програмного забезпечення до системи ERP, від визначення функціоналу для перенесення або модернізації до реалізації, тестування та інтеграції нової системи з існуючими модулями ERP. Наведені приклади роботи розроблених програм, такі як картка коду ЗТО, створення заявки та замовлення на виготовлення ЗТО, сформований план виробництва.

Здійснення всіх вказаних завдань було спрямоване на досягнення важливої мети – інтеграцію системи планування виробництва інструменту та оснащення у основну ERP-систему підприємства. Це оновлене планування сприятиме покращенню ефективності виробництва, підвищенню інтеграції між відділами та загальної ефективності управління.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1 Toomey J. What Is ERP? [Електронний ресурс] / Jennifer Toomey. – 2020. URL: <https://www.oracle.com/erp/what-is-erp/> (last accessed:26.09.2023).

2 Perkins B. What is ERP? Key features of top enterprise resource planning systems [Електронний ресурс] / Bart Perkins. – 2020. URL: <https://www.cio.com/article/272362/what-is-erp-key-features-of-top-enterprise-resource-planning-systems.html> (last accessed:26.09.2023).

3 Oracle Enterprise Resource Planning (ERP). URL: <https://www.oracle.com/ua/erp> (дата звернення: 27.09.2023).

4 Microsoft Dynamics 365. URL:<https://dynamics.microsoft.com/ua-ua/erp/erp-software> (дата звернення: 28.09.2023).

5 SAP /Головна сторінка. URL: https://www.sap.com/ukraine/index.html?url_id=auto_hp_redirect_ukraine (дата звернення: 28.09.2023).

6 Vivek Kale. Implementing SAP R/3: The Guide for Business and Technology Managers. Carmel: Sams Publishing; First Edition, 2000. 528 Pages.

7 Application Development on AS ABAP. URL: https://help.sap.com/docs/SAP_NETWEAVER_750/7bfe8cdcfbb040dcb6702dada8c3e2f0/2ff82a005ddd4f369b74bfda71f297c0.html?version=7.5.9 (last accessed:13.10.2023).

8 Olaf Schulz Using SAP: A Guide for Beginners and End Users. Galileo Press, 2014. 392 Pages.

Додаток А

Лістинг А.1 – Частковий лістинг програми внесення даних у каталог ЗТО

```

REPORT ZPPSxxx1 MESSAGE-ID ZPS1.
TABLES: ZTPSxxx1, ZTPSxxx2, ZTPSxxx3, ZTPSxxx4, ZTPSxxx5,
ZTPSxxx6, ZTPSxxx8, ZTPSxxx9, ZTPSxx14, ZTPSxxx7, AUFK, ZTPSxx15_1,
ZTPSxx15_2, ZTPSxx15_3, ZTPSxx16.
*визначаємо основні структури даних*
DATA: BEGIN OF TABxxx1 OCCURS 0,
      SH LIKE ZTPSxxx1-SH,
      CIZG LIKE ZTPSxxx1-CIZG,
      GR LIKE ZTPSxxx1-GR,
      VO LIKE ZTPSxxx1-VO,
      NAME LIKE ZTPSxxx1-NAME,
      AEDTM LIKE ZTPSxxx1-AEDTM,
      UNAME LIKE ZTPSxxx1-UNAME,
END OF TABxxx1.
* визначаємо додаткові структури даних для тимчасового
збереження даних*
DATA TABxxx1_SAVE LIKE TABLE OF TABxxx1 WITH HEADER LINE.
CONTROLS KARTOTEKA TYPE TABSTRIP.
*controls: osnov type tableview using screen '1101'.
*controls: prorab type tableview using screen '1000'.
PARAMETERS: SHIFR(18) TYPE C NO-DISPLAY.
PARAMETERS: STO(18) TYPE C NO-DISPLAY.
DATA: FCODE LIKE SY-UCOMM, FCODE_V LIKE SY-UCOMM,
      FCODE_OK LIKE SY-UCOMM.
DATA: OSN(1), CIZG(3), GR(2),
      SH LIKE ZTPSxxx1-SH,
      VIDCH LIKE ZTPSxxx1-VIDCH,
      NACHALO VALUE '1',
      PRORABOT VALUE ' ',
      STR_KART(1), KOLCIZGS TYPE I VALUE 0,
      NOMCEX LIKE ZTPSxxx1-CIZG,
      TXTCIS(35),
      LIN TYPE P,
      STR_SH(1),
      STR_OSN(1),
      STR_CIZG(1),
      STR_GR(1),
      DYNNR LIKE SY-DYNNR,
      SHABLON(50),
      NAIM_VO(14),
      NAIM_VO_S(14),
      EDIZ LIKE ZTPSxxx8-EDIZ,
      EDIZ_S LIKE ZTPSxx17-EDIZ_S,
      EDIZ_K LIKE ZTPSxx17-EDIZ_K,
      EDIZ_D LIKE ZTPSxx17-EDIZ_D,
      VO LIKE ZTPSxxx1-VO,

```

```

SELECT_VALUE LIKE SY-DATUM,
DD LIKE SY-DATUM,
CZAK LIKE ZTPSxxx2-CZAK,
SHAB(29),
DATA: PR_SAVE(1)    VALUE ` `,
EXIT_YES_NO  VALUE `0`,
YES_VALUE `1`,
NO    VALUE `0`,
EXIT_KART VALUE `0`,
SYSUBRC LIKE SY-SUBRC,
VVDOD(1),
FLAG(1)    VALUE ` `,
KTEXT LIKE AUFK-KTEXT,
KTEXT_ZAK LIKE AUFK-KTEXT,
CEX_EXIT(1)    VALUE ` `,
USER_ANSWER TYPE T.
DATA: SCR_NAME LIKE SCREEN-NAME,
CUR_NAME LIKE SCREEN-NAME,
POZ_CUR(20).
DATA: CURSOR_LINE LIKE SY-STEPL, CURSOR_FIELD(30).
DATA: EXIT_ZAYAV VALUE `0`,
EXIT_1105 VALUE `0`,
Z_DEL_NODEL VALUE `0`,
VVDOD_ZAK VALUE ` `,
VVDOD_SPLAV(1) VALUE `0`,
DEL_VALUE `1`,
NODEL VALUE `0`,
BUTT TYPE I,
SW_SAVE VALUE `0`.
DATA: BEGIN OF DETERM_IND OCCURS 0,      " визначення полів
IND LIKE SY-INDEX,
END OF DETERM_IND.
DATA: TNAM LIKE HELP_INFO-TABNAME,
FNAM LIKE HELP_INFO-FIELDNAME.
DATA: BEGIN OF IHELP_FIELDS OCCURS 20.
INCLUDE STRUCTURE HELP_VALUE.
DATA: END OF IHELP_FIELDS.
DATA: BEGIN OF HELPTAB OCCURS 100,
FELD(40),
END OF HELPTAB.
DATA: CUCOL LIKE SY-CUCOL VALUE 10,
CUROW LIKE SY-CUROW VALUE 10.
DATA BLOCK(1) TYPE N.
DATA: TXTLINE1 LIKE SPOP-TEXTLINE1,
TXTLINE2 LIKE SPOP-TEXTLINE2.
INCLUDE ZPPSxxx2.
*start-of-selection.
*if sy-uname <> `CXXXXXXX`.
*    message i016 with `користувачу відмовлено у доступі`.
*    exit.
*endif.
INITIALIZATION.
*perform form_ZTABxxx5.

```

```

PERFORM SPRAV_VO.
PERFORM SPRAV_VIDCH.
PERFORM SPRAV_CIZG.
PERFORM SPRAV_GR.
PERFORM PREDFORM.
PERFORM NZ.
MODIFY TABxxx8 INDEX 1.
CALL SCREEN 1100.
*-----
*           Module   STATUS_1100   OUTPUT
*-----
MODULE STATUS_1100 OUTPUT.
  SET TITLEBAR '1100KART'.
  CASE KARTOTEKA-ACTIVETAB.
    WHEN 'OSNOV'.
      SET PF-STATUS 'STAT1101'.
      DYNMR = 1101.
    WHEN OTHERS.
      KARTOTEKA-ACTIVETAB = 'OSNOV'.
      SET PF-STATUS 'STAT1101'.
      DYNMR = 1101.
  ENDCASE.
ENDMODULE.
*-----
*           Module   USER_COMMAND_1100   INPUT
*-----
MODULE USER_COMMAND_1100 INPUT.
CASE FCODE.
  WHEN 'OUT'.
    FCODE = SPACE.
    PERFORM EXIT_KART.
    IF EXIT_KART EQ YES.
      FLAG = ' '.
      CLEAR: POZ_CUR, CUR_NAME, KARTOTEKA-ACTIVETAB.
      LEAVE PROGRAM.
    ELSE.
      FLAG = 'X'.
      EXIT.
    ENDIF.
  WHEN 'SAVE'.
    IF KARTOTEKA-ACTIVETAB = 'OSNOV'.
      PERFORM SAVEOSN.
  WHEN 'PROV'.
    FCODE = SPACE.
    SHIFR = TABxxx1-SH.
*      export sto shifr to memory id 'prov'.
    SUBMIT ZPPSxxx4S_2 WITH SHIFR = TABxxx1-SH
                        AND RETURN.
*      perform form_prov in program ZPPSxxx4s_2
*                        using ' ' shifr
*                        changing sto.
    IMPORT STO SHIFR FROM MEMORY ID 'prov'.
    TABxxx1-SH = STO.

```

```

        FREE MEMORY.
        FCODE = SPACE.
*       when 'NEW'.
*           perform exit_kart.
*           if exit_kart eq yes.
*               flag = ' '.
*               clear: poz_cur, cur_name.
*               fcode = space.
*               perform cleartab.
*           else.
*               flag = 'X'.
*               exit.
*           endif.
        ELSE.
            MESSAGE I016(PG) WITH 'збережіть дані картки'.
            CLEAR: FCODE.
            FLAG = 'X'.
            LEAVE SCREEN.
        ENDIF.
    ENDIF.
WHEN 'OSNOV'
    IF ( TABxxx1-SH IS INITIAL ) AND
        ( TABxxx1-CIZG IS INITIAL ) AND
        ( TABxxx1-GR IS INITIAL ).
        MESSAGE I016(PG) WITH 'Заповніть обов'язкові
поля'.

        FLAG = 'X'.
        EXIT.
    ELSE.
        PERFORM EXIT_KART.
        IF EXIT_KART EQ YES.
            FLAG = ' '.
            KARTOTEKA-ACTIVETAB = FCODE.
            FCODE = SPACE.
        ELSE.
            FLAG = 'X'.
            EXIT.
        ENDIF.
    ENDIF.
WHEN OTHERS.
    CLEAR: FCODE.
    FLAG = 'X'.
    LEAVE SCREEN.
ENDCASE.
ENDMODULE.
*-----
*           module status_1101 OUTPUT
*-----
MODULE STATUS_1101 OUTPUT.
IF FLAG = ' '.
    CLEAR POZ_CUR.
    SET CURSOR FIELD 'TABxxx1-SH'.
ELSE.

```

```

      SET CURSOR FIELD CUR_NAME LINE POZ_CUR.
ENDIF.
*loop at screen.
*endloop.
ENDMODULE.
*-----
*          FORM cleartab.
*-----
FORM CLEARTAB.
REFRESH TABxxx1. CLEAR TABxxx1.
REFRESH TABxxx1_SAVE. CLEAR TABxxx1_SAVE.
*refresh TABxxx2. clear TABxxx2.
*refresh TABxxx2_save. clear TABxxx2_save.
REFRESH TABxxx3. CLEAR TABxxx3.
REFRESH TABxxx3_SAVE. CLEAR TABxxx3_SAVE.
REFRESH TABxxx6. CLEAR TABxxx6.
REFRESH TABxxx6_SAVE. CLEAR TABxxx6_SAVE.
REFRESH ZTABxxx6. CLEAR ZTABxxx6.
REFRESH ZTABxxx6_SAVE. CLEAR ZTABxxx6_SAVE.
*refresh ZTABxxx5. clear ZTABxxx5.
REFRESH TABxxx5. CLEAR TABxxx5.
REFRESH TABxxx5_SAVE. CLEAR TABxxx5_SAVE.
REFRESH TABxxx8. CLEAR TABxxx8.
REFRESH TABxxx8_SAVE. CLEAR TABxxx8_SAVE.
REFRESH TABxx14. CLEAR TABxx14.
REFRESH TABxx14_SAVE. CLEAR TABxx14_SAVE.
CLEAR: NAIM_VO, NAIM_VO_S.
PERFORM PREDFORM.
PERFORM NZ.
MODIFY TABxxx8 INDEX 1.
*perform form_zTABxxx5.
ENDFORM.
COMMIT WORK.
MESSAGE I016(PG) WITH 'Записи введені'.
FLAG = ' '.
CLEAR: POZ_CUR, CUR_NAME.
PERFORM CLEARTAB.
SET SCREEN 0.
LEAVE SCREEN.
ENDFORM.

```

Додаток Б

Лістинг Б.1 – Частковий лістинг програми ведення виробничої бази

```

REPORT ZPPSxxx2 MESSAGE-ID ZPS1.
*оголошення таблиць та структур
TABLES: ZTPSxxx1, ZTPSxxx2, ZTPSxxx3, ZTPSxxx4, ZTPSxxx5,
ZTPSxxx6, ZTPSxxx8, ZTPSxxx14, ZTPSxxx7, AUFK, ZTPSxxx9, ZTPSxxx10,
ZTPSxxx16, ZTPSxxx11, ZTPSxxx12, ZTPSxxx13, ZTPSxxx15_1, ZTPSxxx15_2,
ZTPSxxx15_3.
DATA: BEGIN OF TABxxx1 OCCURS 0,
      SH LIKE ZTPSxxx1-SH,
      CIZG LIKE ZTPSxxx1-CIZG,
      GR LIKE ZTPSxxx1-GR,
      VO LIKE ZTPSxxx1-VO,
      NAME LIKE ZTPSxxx1-NAME,
      AEDTM LIKE ZTPSxxx1-AEDTM,
      UNAME LIKE ZTPSxxx1-UNAME,
END OF TABxxx1.
DATA TABxxx1_SAVE LIKE TABLE OF TABxxx1 WITH HEADER LINE.
DATA: BEGIN OF ZTABxxx2 OCCURS 0,
      SH LIKE ZTPSxxx2-SH,
      CZAK LIKE ZTPSxxx2-CZAK,
      NZAK LIKE ZTPSxxx2-NZAK,
      NDET LIKE ZTPSxxx2-NDET,
      OPERATION LIKE ZTPSxxx2-OPERATION,
END OF ZTABxxx2.
*data TABxxx2_save like table of TABxxx2 with header line.
*data zTABxxx2 like table of TABxxx2 with header line.
DATA: BEGIN OF TABxxx8 OCCURS 0,
      SH LIKE ZTPSxxx8-SH,
      CIZG LIKE ZTPSxxx8-CIZG,
      GR LIKE ZTPSxxx8-GR,
      NZ LIKE ZTPSxxx8-NZ,
      NMER LIKE ZTPSxxx8-NMER,
      NPM LIKE ZTPSxxx8-NPM,
      SRIZG LIKE ZTPSxxx8-SRIZG,
      DPZ LIKE ZTPSxxx8-DPZ,
      DOZ LIKE ZTPSxxx8-DOZ,
      KOL LIKE ZTPSxxx8-KOL,
      EDIZ LIKE ZTPSxxx8-EDIZ,
*      ediz_zak like ZTPSxxx8-ediz_zak,
      NZAK LIKE ZTPSxxx8-NZAK,
      NZAK_R LIKE ZTPSxxx8-NZAK_R,
      VZ LIKE ZTPSxxx8-VZ,
      TR LIKE ZTPSxxx8-TR,
      CZAK LIKE ZTPSxxx8-CZAK,
      NOZ LIKE ZTPSxxx8-NOZ,
      AEDTM LIKE ZTPSxxx8-AEDTM,
      UNAME LIKE ZTPSxxx8-UNAME,

```

```

END OF TABxxx8.
DATA TABxxx8_SAVE LIKE TABLE OF TABxxx8 WITH HEADER LINE.
. . .
DATA: BEGIN OF TABxx14 OCCURS 0,
      NMER LIKE ZTPSxxx14-NMER,
      NAZVM LIKE ZTPSxxx14-NAZVM,
      TD LIKE ZTPSxxx14-TD,
      DPD LIKE ZTPSxxx14-DPD,
      KRK LIKE ZTPSxxx14-KRK,
END OF TABxx14.
DATA TABxx14_SAVE LIKE TABLE OF TABxx14 WITH HEADER LINE.
DATA: BEGIN OF TABxxx9 OCCURS 0,
      SH LIKE ZTPSxxx9-SH,
      CIZG LIKE ZTPSxxx9-CIZG,
      GR LIKE ZTPSxxx9-GR,
      VZ LIKE ZTPSxxx9-VZ,
      DOZ LIKE ZTPSxxx9-DOZ,
      SRIZG LIKE ZTPSxxx9-SRIZG,
      KOL LIKE ZTPSxxx9-KOL,
      CZAK LIKE ZTPSxxx9-CZAK,
      NOZ LIKE ZTPSxxx9-NOZ,
*      nzak_r like ZTPSxxx9-nzak_r,
      DTO LIKE ZTPSxxx9-DTO,
      DZO LIKE ZTPSxxx9-DZO,
      DSS LIKE ZTPSxxx9-DSS,
      DZOF LIKE ZTPSxxx9-DZOF,
      DGR LIKE ZTPSxxx9-DGR,
      EDIZ LIKE ZTPSxxx9-EDIZ,
      KGR LIKE ZTPSxxx9-KGR,
      DPTS LIKE ZTPSxxx9-DPTS,
      EDIZ_GR LIKE ZTPSxxx9-EDIZ,
      PROG LIKE ZTPSxxx9-PROG,
      PROG1 LIKE ZTPSxxx9-PROG1,
*      pplan like ZTPSxxx9-pplan,
      KPLAN LIKE ZTPSxxx9-KPLAN,
*      osttrud like ZTPSxxx9-osttrud,
*      dgrdop like ZTPSxxx9-dgrdop,
      DSD1 LIKE ZTPSxxx9-DSD1,
      DSD2 LIKE ZTPSxxx9-DSD2,
      DSD3 LIKE ZTPSxxx9-DSD3,
      DSD4 LIKE ZTPSxxx9-DSD4,
      KSD1 LIKE ZTPSxxx9-KSD1,
      KSD2 LIKE ZTPSxxx9-KSD2,
      KSD3 LIKE ZTPSxxx9-KSD3,
      KSD4 LIKE ZTPSxxx9-KSD4,
      TR1 LIKE ZTPSxxx9-TR1,
      TR2 LIKE ZTPSxxx9-TR2,
      TR3 LIKE ZTPSxxx9-TR3,
      TR4 LIKE ZTPSxxx9-TR4,
      TRZAK LIKE ZTPSxxx9-TRZAK,
      ND LIKE ZTPSxxx9-ND,
      NZAK LIKE ZTPSxxx9-NZAK,
      PRIM LIKE ZTPSxxx9-PRIM,

```

```

        NAPMK LIKE ZTPSxxx9-NAPMK,
        GODKZ LIKE ZTPSxxx9-GODKZ,
        BRAK LIKE ZTPSxxx9-BRAK,
        GODVP LIKE ZTPSxxx9-GODVP,
        NMK LIKE ZTPSxxx9-NMK,
        UZ LIKE ZTPSxxx9-UZ,
        KOLTA LIKE ZTPSxxx9-KOLTA,
        UNAME LIKE ZTPSxxx9-UNAME,
        AEDTM LIKE ZTPSxxx9-AEDTM,
END OF TABxxx9.
DATA TABxxx9_SAVE LIKE TABLE OF TABxxx9 WITH HEADER LINE.
. . .
PARAMETERS: SHIFR(18) TYPE C NO-DISPLAY.
PARAMETERS: STO(18) TYPE C NO-DISPLAY.
CONTROLS KARTOTEKA TYPE TABSTRIP.
CONTROLS: TABLSH TYPE TABLEVIEW USING SCREEN '0001'.
CONTROLS: KART TYPE TABLEVIEW USING SCREEN '3101'.
CONTROLS: ZAGOT TYPE TABLEVIEW USING SCREEN '1102'.
*controls: vnod type tableview using screen '1103'.
CONTROLS: ZAYAV TYPE TABLEVIEW USING SCREEN '3104'.
CONTROLS: ZAKAZ TYPE TABLEVIEW USING SCREEN '3105'.
CONTROLS: ARHIV TYPE TABLEVIEW USING SCREEN '3106'.
CONTROLS: ANNZAP TYPE TABLEVIEW USING SCREEN '3111'.
CONTROLS: TABLSHAN TYPE TABLEVIEW USING SCREEN '3112'.
CONTROLS: ANNZAYAV TYPE TABLEVIEW USING SCREEN '3114'.
DATA: FCODE LIKE SY-UCOMM, FCODE_MENU LIKE SY-UCOMM,
      FCODE_OK LIKE SY-UCOMM.
DATA: BEGIN OF DETERM_IND OCCURS 0,      " визначення поля
      IND LIKE SY-INDEX,
END OF DETERM_IND.
DATA: BEGIN OF SPOPLI_TAB OCCURS 5.
      INCLUDE STRUCTURE SPOPLI.
DATA: END OF SPOPLI_TAB.
. . .
*-----
*          f_table_kart
*-----
FORM F_TABLE_KART.
CLEAR: TABxxx1, ZTABxxx2, TABxxx3, TABxxx5, TABxxx6, ZTABxxx6.
REFRESH: TABxxx1, ZTABxxx2, TABxxx3, TABxxx5, TABxxx6,
ZTABxxx6.
SELECT SINGLE * FROM ZTPSxxx1
      WHERE SH EQ SH AND
      CIZG EQ CIZG AND
      GR EQ GR.
IF SY-SUBRC EQ 0.
  MOVE-CORRESPONDING ZTPSxxx1 TO TABxxx1.
  APPEND TABxxx1.
  CLEAR TABxxx1_SAVE. REFRESH TABxxx1_SAVE.
  TABxxx1_SAVE[] = TABxxx1[.]*
PERFORM F_TABLE_2028.
SELECT SINGLE * FROM ZTPSxxx3
      WHERE SH EQ SH.

```

```

IF SY-SUBRC EQ 0.
    MOVE-CORRESPONDING ZTPSxxx3 TO TABxxx3.
ENDIF.
APPEND TABxxx3.
CLEAR TABxxx3_SAVE. REFRESH TABxxx3_SAVE.
TABxxx3_SAVE[] = TABxxx3[].
SELECT SINGLE * FROM ZTPSxxx5
    WHERE SH EQ SH.
IF SY-SUBRC EQ 0.
    MOVE-CORRESPONDING ZTPSxxx5 TO TABxxx5.
ENDIF.
APPEND TABxxx5.
CLEAR TABxxx5_SAVE. REFRESH TABxxx5_SAVE.
TABxxx5_SAVE[] = TABxxx5[].
SELECT * FROM ZTPSxxx6
    WHERE SH EQ SH AND
    CIZG EQ CIZG AND
    GR EQ GR.
    IF SY-SUBRC EQ 0.
        MOVE-CORRESPONDING ZTPSxxx6 TO ZTABxxx6.
    ENDIF.
    APPEND ZTABxxx6.
ENDSELECT.
CLEAR ZTABxxx6_SAVE. REFRESH ZTABxxx6_SAVE.
ZTABxxx6_SAVE[] = ZTABxxx6[].
ENDIF.
CLEAR TABxxx6_SAVE. REFRESH TABxxx6_SAVE.
READ TABLE SELE_OSN WITH KEY VO = TABxxx1-VO.
IF SY-SUBRC = 0.
    NAIM_VO = SELE_OSN-NAIM_VO.
    NAIM_VO_S = NAIM_VO.
ENDIF.
ENDFORM.

. . .
*створення нової заявки
*-----
*          form new_zayav1.
*-----
FORM NEW_ZAYAV1.
CLEAR: TABxxx8, TABxx14.
REFRESH: TABxxx8, TABxx14.
CLEAR: NAIM_VZ, NAIM_TD, SHABLON, NAIM_KRK, NZAK_SEEK, IZDELIE.
TABxxx8-SH = TABxxx1-SH.
TABxxx8-CIZG = TABxxx1-CIZG.
TABxxx8-GR = TABxxx1-GR.
TABxxx8-DOZ = SY-DATUM.
PERFORM NZ.
APPEND TABxxx8.
CLEAR TABxxx8_SAVE. REFRESH TABxxx8_SAVE.
TABxxx8_SAVE[] = TABxxx8[].
ENDFORM.

```

```

*збереження нової заявки
*-----
*           form save_zayav1
*-----
FORM SAVE_ZAYAV1.
PERFORM SAVE_TD.
PERFORM F_NMER.
CLEAR: EDIZ, TXT_EDIZ, FCODE.
CLEAR: SW_SAVE, SYSUBRC.
PERFORM PROV_VZ USING NAIM_VZ.
IF SYSUBRC NE 0.
    SET CURSOR FIELD 'TABxxx8-VZ'.
    EXIT.
ENDIF.
IF TABxxx8-DOZ EQ 0.
    MESSAGE I016(ZPS1) WITH 'вказіть дату відкриття заявки'.
    SET CURSOR FIELD 'TABxxx8-DOZ'.
    EXIT.
ENDIF.
IF TABxxx14-DPD EQ 0.
    MESSAGE I016(ZPS1) WITH 'вказіть дату підписання документу'.
    SET CURSOR FIELD 'TABxxx14-DPD'.
    EXIT.
ENDIF.
PERFORM PROV_TD.
IF SYSUBRC NE 0.
    SET CURSOR FIELD 'NAIM_TD'.
    EXIT.
ENDIF.
PERFORM PROV_NAZVM.
IF SYSUBRC NE 0.
    SET CURSOR FIELD 'TABxxx8-NMER'.
    EXIT.
ENDIF.
PERFORM PROV_NZAK.
IF SYSUBRC NE 0.
    SET CURSOR FIELD 'TABxxx8-NZAK'.
    EXIT.
ENDIF.
IF TABxxx8-SRIZG EQ 0.
    MESSAGE I016(ZPS1) WITH 'вказіть термін виготовлення'.
    SET CURSOR FIELD 'TABxxx8-SRIZG'.
    EXIT.
ENDIF.
IF TABxxx8-KOL NE 0.
    EDIZ = TABxxx8-EDIZ.
    TXT_EDIZ = 'заявка'.
    PERFORM PROV_EDIZ.
    IF SYSUBRC NE 0.
        SET CURSOR FIELD 'TABxxx8-EDIZ_NZAK'.
        EXIT.
    ENDIF.
ENDIF.
ENDIF.

```

```

PERFORM PROV_CZAK.
IF SYSUBRC NE 0.
    SET CURSOR FIELD 'TABxxx8-NZAK'.
    EXIT.
ENDIF.
*perform save_td.
ZTPSxx14-TD = TABxx14-TD.
MODIFY TABxx14 INDEX 1.
ZTPSxx14-NMER = TABxx14-NMER.
ZTPSxx14-NAZVM = TABxx14-NAZVM.
ZTPSxx14-TD = TABxx14-TD.
ZTPSxx14-DPD = TABxx14-DPD.
ZTPSxx14-KRK = TABxx14-KRK.
MODIFY ZTPSxx14.
CLEAR TABxx14_SAVE. REFRESH TABxx14_SAVE.
TABxx14_SAVE[] = TABxx14[].
TABxxx8-SH = TABxxx1-SH.
TABxxx8-CIZG = TABxxx1-CIZG.
TABxxx8-GR = TABxxx1-GR.
TABxxx8-NMER = TABxx14-NMER.
CLEAR TR.
TR = TABxxx1-TR * TABxxx8-KOL.
TABxxx8-TR = TR.
PERFORM SAVE_VZ.
TABxxx8-UNAME = SY-UNAME.
TABxxx8-AEDTM = SY-DATUM.
MODIFY TABxxx8 INDEX 1.
ZTPSxxx8-SH = TABxxx8-SH.
ZTPSxxx8-NMER = TABxxx8-NMER.
ZTPSxxx8-NPM = TABxxx8-NPM.
ZTPSxxx8-SRIZG = TABxxx8-SRIZG.
ZTPSxxx8-KOL = TABxxx8-KOL.
ZTPSxxx8-EDIZ = TABxxx8-EDIZ.
ZTPSxxx8-NZAK = TABxxx8-NZAK.
*ZTPSxxx8-nzak_r = TABxxx8-nzak_r.
ZTPSxxx8-VZ = TABxxx8-VZ.
ZTPSxxx8-TR = TABxxx8-TR.
ZTPSxxx8-CIZG = TABxxx8-CIZG.
ZTPSxxx8-GR = TABxxx8-GR.
ZTPSxxx8-NZ = TABxxx8-NZ.
ZTPSxxx8-CZAK = TABxxx8-CZAK.
ZTPSxxx8-NOZ = TABxxx8-NOZ.
ZTPSxxx8-NDET = TABxxx8-NDET.
ZTPSxxx8-OPERATION = TABxxx8-OPERATION.
ZTPSxxx8-DOZ = TABxxx8-DOZ.
ZTPSxxx8-DPZ = TABxxx8-DPZ.
ZTPSxxx8-PRW2 = TABxxx8-PRW2.
ZTPSxxx8-PRW3 = TABxxx8-PRW3.
ZTPSxxx8-PRW4 = TABxxx8-PRW4.
ZTPSxxx8-UNAME = SY-UNAME.
ZTPSxxx8-AEDTM = SY-DATUM.
ZTPSxxx8-AETIM = SY-UZEIT.
MODIFY ZTPSxxx8.

```

```
CLEAR TABxxx8_SAVE. REFRESH TABxxx8_SAVE.
TABxxx8_SAVE[] = TABxxx8[].
NAIM_TD_S = NAIM_TD. NAIM_VZ_S = NAIM_VZ.NAIM_KRK_S = NAIM_KRK.
SELECT SINGLE * FROM ZTPSxxx2
    WHERE SH EQ TABxxx8-SH AND
    CZAK EQ TABxxx8-CZAK AND
    NZAK EQ TABxxx8-NZAK AND
    NDET EQ TABxxx8-NDET AND
    OPERATION EQ TABxxx8-OPERATION.
COMMIT WORK.
MESSAGE I016(PG) WITH 'Дані введені'.
FLAG = ' '.
CLEAR: POZ_CUR, CUR_NAME.
IF KARTOTEKA-ACTIVETAB = 'ZAYAV'.
ELSE.
SET SCREEN 0.
LEAVE SCREEN.
ENDIF.
ENDLOOP.
ENDFORM.
```

Додаток В

Лістинг В.1 – Частковий лістинг програми формування плану виробництва

```

REPORT ZPPSxxx3  message-id zps1.
TABLES: T247,      " повні та короткі назви місяців
        ZTPSxxx1,  " каталог ЗТО (KARTOSN)
        ZTPSxxx2,  " підрозділи замовники
        ZTPSxxx4,  " норми витрат (KARTLIM)
        ZTPSxxx8,  " облік первинних замовлень (PERV)
        ZTPSxx18,  " облік первинних замовлень додатки (DPERV)
        ZTPSxxx9,  " виробничі замовлення (KZAK)
        ZTPSxx10,  " архів (ARXIZG)
        ZTPSxx19,  " дефіцит (DEFIC)
        ZTPSxx20,  " перелік підрозділів виробництва
        ZTPSxx21.  " план випуску
*          BUCIO1   " дані складу модуль MM
*          BUCIRAS  " потужність модуль HR
DATA: ok_code LIKE sy-ucomm,
      save_ok LIKE sy-ucomm,
      wcizg    LIKE ZTPSxx20-izgotov,
      ceh      LIKE ZTPSxx20-cizg,
      grup     LIKE ZTPSxx20-gr,
      nm       LIKE ZTPSxx21-ms,
      wyear    LIKE ZTPSxx21-gd,
      wsh      LIKE ZTPSxxx8-sh,
      wsh1     LIKE ZTPSxxx8-sh,
      wsh2     LIKE ZTPSxxx8-sh,
      psh      LIKE ZTPSxxx1-sh,
      name_mes(10),
      wDATE(10),
      kzaya(3) TYPE n,
      kzakaz(3) TYPE n,
      karhiv(3) TYPE n,
      l_answer(1),
      r1(1), r2(1),
      kk(3) TYPE n VALUE 100, " розмір набору плану у кодах ЗТО
      wtext2(90),

. . .
DATA:
      application type ole2_object,
      sheet type ole2_object,
*      cells type ole2_object,
*      m_cells type ole2_object,
*      selection type ole2_object,
      workbooks type ole2_object,
      ROWS TYPE OLE2_OBJECT,
*      l_pgst type ole2_object,
*      m_column type ole2_object,
      m_range type ole2_object,

```

```

        mec_range type ole2_object,
*       m_inter type ole2_object,
        m_fnt type ole2_object,
        borders type ole2_object.
*       workbook type ole2_object.
. . .
TYPES: BEGIN OF t_plan,
        sh      LIKE ZTPSxxx8-sh,           " PERV
        nmer    LIKE ZTPSxxx8-nmer,
        npm     LIKE ZTPSxxx8-npm,
        vz      LIKE ZTPSxxx8-vz,
        czak    LIKE ZTPSxxx8-czak,
        nzak    LIKE ZTPSxxx8-nzak,
        srizg   LIKE ZTPSxxx8-srizg,
        kzak    LIKE ZTPSxxx8-kzak,
        ediz_zak LIKE ZTPSxxx9-ediz_zak,
        tred    LIKE ZTPSxxx8-tred,
        noz     LIKE ZTPSxxx8-noz,
        krk     LIKE ZTPSxxx8-krk,
        kksh    LIKE ZTPSxxx8-kksh,
        ediz_ksh LIKE ZTPSxxx8-ediz_zak,
        kvs     LIKE ZTPSxx18-kvs,
        ediz_s  LIKE ZTPSxx18-ediz_s,
        kvk     LIKE ZTPSxx18-kvk,
        ediz_k  LIKE ZTPSxx18-ediz_k,
        kvd     LIKE ZTPSxx18-kvd,
        ediz_d  LIKE ZTPSxx18-ediz_d,
        dto     LIKE ZTPSxxx9-dto,
        dzo     LIKE ZTPSxxx9-dzo,
        dss     LIKE ZTPSxxx9-dss,
        dgr     LIKE ZTPSxxx9-dgr,
        mark(1),
        END OF t_plan.
. . .
DATA: ltab_fields LIKE help_value OCCURS 0 WITH HEADER LINE,
      selectfield LIKE help_info-fieldname,
      selectval   LIKE help_info-fldvalue.
DATA: BEGIN OF ltab_values OCCURS 0,
        field(40),
      END OF ltab_values,
      BEGIN OF full_table OCCURS 0,
        nmes(2) TYPE n,
      END OF full_table,
      BEGIN OF t_vibzak OCCURS 0,           "
        nzak LIKE ZTPSxxx1-nzak,          " char(6)
      END OF t_vibzak,
      BEGIN OF t_noz OCCURS 0,
        noz LIKE ZTPSxxx8-noz,
      END OF t_noz,
      BEGIN OF krperv OCCURS 0,
        sh      LIKE ZTPSxxx8-sh,
        nmer    LIKE ZTPSxxx8-nmer,
        npm     LIKE ZTPSxxx8-npm,

```

```

        vz      LIKE ZTPSxxx8-vz,
        czak    LIKE ZTPSxxx8-czak,
        nzak    LIKE ZTPSxxx8-nzak,
        srizg   LIKE ZTPSxxx8-srizg,
        kzak    LIKE ZTPSxxx8-kzak,
        ediz_zak LIKE ZTPSxxx8-ediz_zak,
        kksh    LIKE ZTPSxxx8-kksh,
        ediz_ksh LIKE ZTPSxxx8-ediz_ksh,
        tred    LIKE ZTPSxxx8-tred,
        noz     LIKE ZTPSxxx8-noz,
        END OF krperv,
**** тимчасово створюю таблицю. PLANVIP (потім ZTPSxx21) план
випуску
*      BEGIN OF planzap OCCURS 0,
*      sh      LIKE ZTPSxxx8-sh,          " PERV
*      nmer    LIKE ZTPSxxx8-nmer,
*      npm     LIKE ZTPSxxx8-npm,
*      vz      LIKE ZTPSxxx8-vz,
*      czak    LIKE ZTPSxxx8-czak,
*      nzak    LIKE ZTPSxxx8-nzak,
*      srizg   LIKE ZTPSxxx8-srizg,
*      kzak    LIKE ZTPSxxx8-kzak,
*      tred    LIKE ZTPSxxx8-tred,
*      noz     LIKE ZTPSxxx8-noz,
*      krk     LIKE ZTPSxxx8-krk,
*      kksh    LIKE ZTPSxxx8-kksh,
*      kvs     LIKE ZTPSxxx18-kvs,
*      ediz_s  LIKE ZTPSxxx18-ediz_s,
*      kvk     LIKE ZTPSxxx18-kvk,
*      ediz_k  LIKE ZTPSxxx18-ediz_k,
*      kvd     LIKE ZTPSxxx18-kvd,
*      ediz_d  LIKE ZTPSxxx18-ediz_d,
*      mark(1),
*      END OF planzap,
. . .
* тимчасова таблиця для повторного включення у план випуску
* потім ZTPSxx21 vpl = 'П'
      BEGIN OF repeat OCCURS 0,
        sh      LIKE ZTPSxxx9-sh,
        name    LIKE ZTPSxxx1-name,
        doz     LIKE ZTPSxxx9-doz,
        srizg   LIKE ZTPSxxx9-srizg,
        noz     LIKE ZTPSxxx9-noz,
        mark(1),
      END OF repeat,
* заявки на виготовлення ЗТО
      BEGIN OF st_perv OCCURS 0,
        sh      LIKE ZTPSxxx8-sh,
        nmer    LIKE ZTPSxxx8-nmer,
        npm     LIKE ZTPSxxx8-npm,
        srizg   LIKE ZTPSxxx8-srizg,
        vz      LIKE ZTPSxxx8-vz,
        czak    LIKE ZTPSxxx8-czak,

```

```

        nzak    LIKE ZTPSxxx8-nzak,
        kzak    LIKE ZTPSxxx8-kzak,
ediz_zak LIKE ZTPSxxx8-ediz_zak,
        tred    LIKE ZTPSxxx8-tred,
        noz     LIKE ZTPSxxx8-noz,
        kksh    LIKE ZTPSxxx8-kksh,
ediz_ksh LIKE ZTPSxxx8-ediz_ksh,
        kvs     LIKE ZTPSxxx18-kvs,
        ediz_s  LIKE ZTPSxxx18-ediz_s,
        kvk     LIKE ZTPSxxx18-kvk,
        ediz_k  LIKE ZTPSxxx18-ediz_k,
        kvd     LIKE ZTPSxxx18-kvd,
        ediz_d  LIKE ZTPSxxx18-ediz_d,
        END OF st_perv,
*   замовлення на виготовлення
        BEGIN OF st_kzak OCCURS 0,
            sh    LIKE ZTPSxxx9-sh,
            srizg LIKE ZTPSxxx9-srizg,
            noz    LIKE ZTPSxxx9-noz,
            vz     LIKE ZTPSxxx9-vz,
            nzak   LIKE ZTPSxxx9-nzak,
            czak   LIKE ZTPSxxx9-czak,
            kzak   LIKE ZTPSxxx9-kzak,
ediz_zak LIKE ZTPSxxx9-ediz_zak,
            kksh   LIKE ZTPSxxx9-kksh,
ediz_ksh LIKE ZTPSxxx9-ediz_ksh,
            trzak  LIKE ZTPSxxx9-trzak,
            prog   LIKE ZTPSxxx9-prog,
            dto    LIKE ZTPSxxx9-dto,
            dzo    LIKE ZTPSxxx9-dzo,
            dss    LIKE ZTPSxxx9-dss,
            dgr    LIKE ZTPSxxx9-dgr,
            prim   LIKE ZTPSxxx9-prim,
        END OF st_kzak.
*   для друку плану випуску
DATA: BEGIN OF imbp OCCURS 0.                                " PxxxXXxx.dbf
        INCLUDE TYPE t_plan.
DATA:  xi1    LIKE ZTPSxxx1-xi1,
        xi2    LIKE ZTPSxxx1-xi2,
        prog1  LIKE ZTPSxxx9-prog1,
        gr     LIKE ZTPSxxx9-gr,
        cizg   LIKE ZTPSxxx9-cizg,
        doz    LIKE ZTPSxxx9-doz,
        prog   LIKE ZTPSxxx9-prog,
        dopoln LIKE ZTPSxxx18-dopoln,
        prim(40),
END OF imbp,
BEGIN OF imbz OCCURS 0,                                " ZAKxxxXX.dbf
        sh    LIKE ZTPSxxx9-sh,
        vz    LIKE ZTPSxxx9-vz,
        nzak  LIKE ZTPSxxx9-nzak,
        czak  LIKE ZTPSxxx9-czak,
        kzak  LIKE ZTPSxxx9-kzak,

```

```

    ediz_zak LIKE ZTPSxxx9-ediz_zak,
    kksh    LIKE ZTPSxxx9-kksh,
    ediz_ksh LIKE ZTPSxxx9-ediz_ksh,
    kvs     LIKE ZTPSxx18-kvs,
    ediz_s  LIKE ZTPSxx18-ediz_s,
    kvk     LIKE ZTPSxx18-kvk,
    ediz_k  LIKE ZTPSxx18-ediz_k,
    kvd     LIKE ZTPSxx18-kvd,
    ediz_d  LIKE ZTPSxx18-ediz_d,
    srizg   LIKE ZTPSxxx8-srizg,
    dopl(11),
END OF imbz.

. . .
START-OF-SELECTION.
    CALL SCREEN 100.                                " екран вибору критеріїв
*-----*
*           Module   STATUS_0100   OUTPUT
*-----*
MODULE STATUS_0100 OUTPUT.
    SET PF-STATUS '0100'.
    SET TITLEBAR '0100'.
LOOP AT SCREEN.
    IF screen-name = 'WTEXT1' OR
       screen-name = 'WTEXT5'.
        IF r3 = 'X'. screen-invisible = 0.
        ELSE.       screen-invisible = 1.
        ENDIF.
        MODIFY SCREEN.
    ELSEIF screen-name = 'WTEXT3' OR
           screen-name = 'WTEXT4'.
        IF r5 = 'X'. screen-invisible = 0.
        ELSE.       screen-invisible = 1.
        ENDIF.
        MODIFY SCREEN.
    ELSEIF screen-name = 'WTEXT6'.
        IF tab_vibzak-lines = 0. screen-invisible = 1.
        ELSE.                   screen-invisible = 0.
        ENDIF.
    ENDIF.
ENDLOOP.

IF ok_code = 'ENTER'.
    CASE wfield.
        WHEN 'WCIZG'. SET CURSOR FIELD 'NM'.
        WHEN 'NM'.   SET CURSOR FIELD 'R1'.
        WHEN 'R1' OR 'R2'. SET CURSOR FIELD 'KK'.
        WHEN 'KK'.     SET CURSOR FIELD 'SIMV'.
        WHEN 'SIMV'.   SET CURSOR FIELD 'R3'.
        WHEN 'R3' OR 'R4'. SET CURSOR FIELD 'R5'.
        WHEN 'R5' OR 'R6'. SET CURSOR FIELD 'R7'.
        WHEN 'R7' OR 'R8'. SET CURSOR FIELD 'R7'.
    ENDCASE.
ENDIF.

```

```

ENDMODULE.                                " STATUS_0100  OUTPUT
. . .
**План випуску зберігається тільки після виводу на друк!
*   CLEAR t_plvip. REFRESH t_plvip.
*   LOOP AT planzap.                        " випуск
*       MOVE-CORRESPONDING planzap TO t_plvip.
*       t_plvip-gd   = wyear.
*       t_plvip-ms   = nm.
*       t_plvip-cizg = ceh.
*       t_plvip-gr   = grup+1(1).
*       t_plvip-vpl  = '3'.
**       APPEND t_plvip.
*       MODIFY ZTPSxx21 FROM t_plvip.
*   ENDLOOP.
*   LOOP AT plandef.
*       MOVE-CORRESPONDING plandef TO t_plvip.
*       t_plvip-gd   = wyear.
*       t_plvip-ms   = nm.
*       t_plvip-cizg = ceh.
*       t_plvip-gr   = grup+1(1).
*       t_plvip-vpl  = 'Д'.
**       APPEND t_plvip.
*       MODIFY ZTPSxx21 FROM t_plvip.
*   ENDLOOP.
. . .
***>>> Результати складання плану
PERFORM Result.
CALL SCREEN 0108 STARTING AT 50 1
                                ENDING AT 200 30. " кількість обраних шифрів
*   ELSE.                        " набрана кількість кодів >= бажаного
*   ENDIF.
. . .
FORM Settings1 USING t_num1 t_num2 val lin colmwidth stlb.
CALL METHOD OF sheet 'Range' = m_range NO FLUSH
    exporting #1 = t_num1    #2 = t_num2.
IF lin = 2 and stlb = 8.        "val = 'Стан у виробництві'.
    set property of m_range 'MergeCells' = 1 NO FLUSH. "поєднує
поля
ELSE.
    set property of m_range 'mergecells' = 0 NO FLUSH.
ENDIF.
    set property of m_range 'RowHeight'    = 12 NO FLUSH.
    set property of m_range 'ColumnWidth'  = colmwidth NO FLUSH.
* set property of m_range 'ShrinkToFit'    = 0 NO FLUSH.
    set property of m_range 'WrapText'     = 1 NO FLUSH.
    set property of m_range 'VerticalAlignment' = 2 NO FLUSH.
    set property of m_range 'HorizontalAlignment' = 3 NO FLUSH.
    call method of m_range 'Font' = m_fnt NO FLUSH.
    set property of m_fnt 'Color' = 5 NO FLUSH.
    set property of m_fnt 'Size' = 11 NO FLUSH.
    set property of m_fnt 'Name' = 'Times New Roman CYR' NO
FLUSH.
    set property of m_fnt 'Bold' = 0 NO FLUSH.

```

```

CALL METHOD OF sheet 'Cells' = mec_range NO FLUSH
    exporting #1 = lin      #2 = stlb.
set property of mec_range 'Value' = val NO FLUSH.
FORM Errors.
    IF sy-subrc ne 0.
        WRITE :/ 'Error зв'язку с EXCEL' , sy-msgli.
        EXIT.
    ENDIF.
ENDFORM.
. . .

```