

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ВОДНОГО ГОСПОДАРСТВА ТА ПРИРОДОКОРИСТУВАННЯ**  
Навчально-науковий інститут  
кібернетики, інформаційних технологій та інженерії  
**Кафедра комп'ютерних технологій та економічної кібернетики**

**Допущено до захисту:**  
Завідувач кафедри  
комп'ютерних технологій та економічної  
кібернетики  
д. е. н., проф. П. М. Грицюк

« \_\_\_\_\_ » \_\_\_\_\_ 2024 р.

**КВАЛІФІКАЦІЙНА РОБОТА**

«Удосконалення інформаційної системи бухгалтерського та  
управлінського обліку НУВГП на основі ISpro»

Виконав:  
студент спеціальності 126  
«Інформаційні технології в бізнесі»  
VI курсу групи ІТБ-61м  
Омелянюк Андрій Андрійович  
Керівник:  
к.т.н. доцент, Василів В.Б.  
Рецензент:  
Волошин В.С.

Рівне – 2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний університет водного господарства та природокористування  
Навчально-науковий інститут кібернетики інформаційних технологій та  
інженерії  
Кафедра комп'ютерних технологій та економічної кібернетики

Освітньо-кваліфікаційний рівень - магістр  
Освітньо-професійна програма «Інформаційні технології в бізнесі»  
Спеціальність 126 «Інформаційні системи і технології»

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
комп'ютерних технологій та  
економічної кібернетики  
д.е.н., професор П.М. Грицюк

“ ” \_\_\_\_\_ 2024 року

**ЗАВДАННЯ**  
**НА КВАЛІФІКАЦІЙНУ РОБОТУ**  
**Омелянук Андрій Андрійович**

1. Тема роботи **Удосконалення інформаційної системи бухгалтерського та управлінського обліку НУВГП на основі ISpro»**

керівник роботи: к.т. н. доцент Василів В.Б.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджена наказом по університету від С№ \_\_\_\_\_ від \_\_\_\_\_ 2024 р.

2. Термін здачі студентом закінченої роботи 20.12.2024 року

3. Вихідні дані до роботи Посібники, наукові статті, інтернет джерела за темою роботи.

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

*Вступ. Теоретичні основи побудови та функціонування інформаційних систем бухгалтерського та управлінського обліку. Інформаційні системи бухгалтерського та управлінського обліку: поняття та класифікація. Особливості побудови інформаційних систем бухгалтерського та управлінського обліку в вищих навчальних закладах. Вимоги до сучасних інформаційних систем обліку в контексті цифрової трансформації. Огляд програмного забезпечення для автоматизації бухгалтерського та управлінського обліку. Переваги та недоліки використання ISpro для автоматизації обліку. Аналіз стану інформаційної системи бухгалтерського та управлінського обліку НУВГП. Загальна характеристика НУВГП та його інформаційної системи. Аналіз бізнес-процесів бухгалтерського та управлінського обліку в НУВГП. Аналіз ефективності функціонування ISpro. Удосконалення інформаційної системи бухгалтерського та управлінського обліку*

*НУВГП на основі ISpro. Стратегія впровадження системи ISpro в НУВГП. Створення модуля електронної розсилки відомостей про заробітну плату ISpro. Розробка функціонального звіту реєстр соціальних пільг студентів на базі ISpro. Розробка модуля API для ISpro. Скрипти користувача ISpro WEB.*

## 5. Перелік графічного матеріалу

*Презентація за матеріалами бакалаврської роботи.*

### 1. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
I - III	Василів В.Б.		

Дата видачі завдання 20.12.2024р.

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Термін виконання етапів (роботи)	Примітка
1	Робота над Розділом 1.	10.10.2024-7.11.2024	
2	Робота над Розділами 2,3	10.11.2024-7.12.2024	
3	Підготовка пояснювальної записки	8.10.2024-14.12.2024	
4	Написання вступу, висновків, реферату	16.12.2024	
5	Попередній захист роботи	17.12.2024	
6	Підготовка презентації роботи	18.12.2024	
7	Відгук керівника, рецензування роботи, перевірка на плагіат	19.12.2024	
8	Допуск до захисту	20.12.2024	

Студент \_\_\_\_\_ Омелянюк А.А.  
(підпис) (прізвище і ініціали)

**Керівник кваліфікаційної роботи**  
\_\_\_\_\_ Василів В.Б.

Зміст	
Вступ.....	6
1 РОЗДІЛ Теоретичні основи побудови та функціонування інформаційних систем бухгалтерського та управлінського обліку .....	9
1.1 Інформаційні системи бухгалтерського та управлінського обліку: поняття та класифікація.....	9
1.2 Особливості побудови інформаційних систем бухгалтерського та управлінського обліку в вищих навчальних закладах .....	13
1.3 Вимоги до сучасних інформаційних систем обліку в контексті цифрової трансформації .....	17
1.4 Огляд програмного забезпечення для автоматизації бухгалтерського та управлінського обліку.....	19
1.5 Переваги та недоліки використання ISpro для автоматизації обліку	21
2 РОЗДІЛ. Аналіз стану інформаційної системи бухгалтерського та управлінського обліку НУВГП.....	29
2.1 Загальна характеристика НУВГП та його інформаційної системи	29
2.2 Аналіз бізнес-процесів бухгалтерського та управлінського обліку в НУВГП	33
2.3 Аналіз ефективності функціонування ISpro.....	38
3 РОЗДІЛ. Удосконалення інформаційної системи бухгалтерського та управлінського обліку НУВГП на основі ISpro .....	44
3.1 Стратегія впровадження системи ISpro в НУВГП.....	44
3.2 Створення модуля електронної розсилки відомостей про заробітну плату ISpro.....	47
3.3 Розробка функціонального звіту реєстр соціальних пільг студентів на базі ISpro .....	52

3.4	Розробка модуля API для ISpro.....	55
3.5	Скрипти користувача ISpro WEB .....	60
	ВИСНОВКИ.....	66
	ВИКОРИСТАНІ ДЖЕРЕЛА .....	68
	Додатки.....	69

## Вступ

В умовах динамічного розвитку економіки та зростання вимог до ефективності управління, інформаційні технології відіграють ключову роль у діяльності будь-якої організації, зокрема й вищих навчальних закладів (ВНЗ). Сучасні ВНЗ, як складні соціально-економічні системи, потребують оперативного та достовірного інформаційного забезпечення для прийняття обґрунтованих управлінських рішень. Особливого значення набуває автоматизація процесів бухгалтерського та управлінського обліку, що дозволяє оптимізувати використання ресурсів, підвищити прозорість фінансової діяльності та забезпечити своєчасне формування звітності.

Актуальність теми дослідження зумовлена низкою факторів:

Зростання обсягів інформації: Збільшення кількості студентів, наукових проєктів, господарських операцій призводить до значного зростання обсягів облікової інформації, що ускладнює її обробку вручну та підвищує ризик помилок.

Підвищення вимог до оперативності та достовірності інформації: Для ефективного управління ВНЗ необхідно мати оперативний доступ до актуальної та достовірної інформації про фінансовий стан, витрати, доходи та інші ключові показники.

Необхідність забезпечення прозорості та підзвітності: Автоматизовані системи обліку забезпечують прозорість фінансової діяльності ВНЗ та сприяють підвищенню рівня підзвітності перед державними органами та громадськістю.

Впровадження цифрової трансформації в освіті: Сучасні тенденції розвитку освіти передбачають активне використання інформаційних технологій у всіх сферах діяльності ВНЗ, включаючи бухгалтерський та управлінський облік.

Національний університет водного господарства та природокористування (НУВГП), як один із провідних ВНЗ України, також потребує постійного вдосконалення своєї інформаційної системи обліку. Наразі в університеті

використовується система ISpro для автоматизації облікових процесів. Проте, аналіз її функціонування виявив певні недоліки та можливості для покращення.

Мета дипломної роботи полягає в удосконаленні інформаційної системи бухгалтерського та управлінського обліку НУВГП шляхом розробки та впровадження додаткових функціональних можливостей на базі системи ISpro.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

Провести теоретичний аналіз існуючих інформаційних систем бухгалтерського та управлінського обліку, їх класифікацію та особливості застосування у ВНЗ.

Дослідити стан інформаційної системи бухгалтерського та управлінського обліку НУВГП, проаналізувати бізнес-процеси та оцінити ефективність функціонування системи ISpro.

Обґрунтувати вибір системи ISpro як платформи для подальшої автоматизації обліку в НУВГП.

Розробити та описати функціональну модель вдосконаленої системи, включаючи створення модуля електронної розсилки відомостей про заробітну плату та розробку функціонального звіту реєстру соціальних пільг студентів.

Оцінити потенційну ефективність від впровадження запропонованих удосконалень.

Об'єктом дослідження є інформаційна система бухгалтерського та управлінського обліку НУВГП.

Предметом дослідження є процес удосконалення інформаційної системи бухгалтерського та управлінського обліку на основі системи ISpro.

Для вирішення поставлених завдань використано такі методи дослідження: аналіз наукової літератури, нормативних документів, метод системного аналізу, моделювання бізнес-процесів, методи порівняльного аналізу та узагальнення.

Структура роботи: дипломна робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків (за необхідності). Перший розділ присвячено теоретичним основам побудови та функціонування інформаційних систем обліку. У другому розділі проведено аналіз стану інформаційної системи НУВГП. У третьому розділі запропоновано шляхи удосконалення системи обліку на базі ISpro.

Робота спрямована на вирішення актуальної проблеми автоматизації документообігу в установах, що дозволяє підвищити якість обслуговування працівників і студентів, економити ресурси та забезпечувати швидкий доступ до фінансової інформації. Результати роботи мають практичну значущість для оптимізації фінансових процесів та їх інтеграції у сучасну цифрову інфраструктуру організації.

# **1 РОЗДІЛ Теоретичні основи побудови та функціонування інформаційних систем бухгалтерського та управлінського обліку**

## **1.1 Інформаційні системи бухгалтерського та управлінського обліку: поняття та класифікація**

Впровадження електронно-обчислювальних машин (ЕОМ), зокрема табличних і перфораційних, спричинило значні зміни у журнально-ордерній формі бухгалтерського обліку. Наприкінці 1950-х років, із появою перфораційних обчислювальних машин, була створена таблично-перфокартова форма обліку. У цій формі реєстри аналітичного та синтетичного обліку генерувалися у вигляді таблиць-табуляграм на основі уніфікованих первинних, підсумкових і умовно-постійних перфокарт, що виконували роль машинних носіїв інформації.

З розвитком обчислювальної техніки у 1970-х роках, із впровадженням комп'ютерів третього покоління (великих і середніх) та багатотермінальних обчислювальних систем, з'явилася таблично-автоматизована форма обліку. Вона дозволила фіксувати первинні дані не лише на паперових носіях, а й безпосередньо на машинних. Це сприяло автоматизації процесу збору первинної інформації. [2, с. 14], [1, с. 158].

До особливостей цієї форми обліку належить:

- Збереження інформації на машинних носіях: поточна, нормативно-довідкова та вхідна інформація могла зберігатися на електронних носіях, що спрощувало її багаторазове використання.
- Створення баз даних: нормативно-довідкова інформація, яка використовувалася багаторазово, інтегрувалася у спеціальні бази даних.
- Методи введення даних: облікова інформація вводилася безпосередньо або за допомогою периферійних пристроїв.

- Автоматизоване формування звітності: за запитом користувача можна було отримувати звіти за необхідними показниками, що підвищувало оперативність і гнучкість обліку.

Автоматизація бухгалтерського обліку стала можливою завдяки розвитку інформаційних технологій. Ця форма обліку передбачає використання комп'ютерних технологій для збору, зберігання та обробки даних. Терміни, які використовуються для її опису, варіюються: "комп'ютерна" [1, с. 158], "діалогово-автоматизована" [4, с. 13], "діалогова" [6, с. 11], "електронна" [7, с. 13–17.].

Таблиця 1.1

Характеристика автоматизованої системи обліку: [1, с. 168]:

Розподілена обробка облікових даних	Обробка інформації здійснюється на різних етапах із використанням розподілених обчислювальних систем
Автоматизація всіх етапів обліку	Включає збір первинної інформації, накопичення, обробку даних та формування звітності
Використання безпаперових технологій	Первинні облікові дані фіксуються в електронному вигляді, передаються дистанційними каналами зв'язку, що дозволяє автоматично складати бухгалтерські проведення.
Захищеність	Система забезпечує збереження даних, захист від несанкціонованого доступу та їх оперативне виявлення за потребою.
Доступність	Користувачі можуть отримувати інформацію на будь-який момент часу відповідно до запиту.
Моделювання господарських операцій	На основі облікових даних здійснюється аналіз операцій і створення моделей для підтримки прийняття управлінських рішень.
Уникнення дублювання	Формується єдиний підхід до обробки даних, виключаючи дублювання показників

Автоматизовані системи бухгалтерського обліку забезпечують ефективну обробку та зберігання великих обсягів однотипної інформації, наприклад, контактних даних, дозволяючи зручно шукати й вибирати дані за визначеними критеріями, такими як адреси, рахунки або звітність. Вони здатні виконувати складні математичні розрахунки, включаючи побудову графіків чи прогнозів, а також багаторазово відтворювати завдання, наприклад, друкувати документи. Завдяки цим системам можна формувати аналітичні та узагальнюючі звіти на будь-яку дату чи період, швидко адаптувати роботу до змін у правових нормах.

Характерною особливістю облікової інформації є її безперервність, що обумовлена внутрішніми зв'язками елементів обліку та місцем підприємства в економічній системі. Інформація має циклічність, оскільки обробляється періодично, хоча її зміст може змінюватися. Вона також відзначається масовістю, адже включає значний обсяг однотипних арифметичних операцій, та суттєвістю, яка концентрує увагу на значущих факторах. У разі складних розрахунків автоматизована система стає незамінною, адже допомагає оптимізувати обробку великої кількості логічних і арифметичних дій.

Системи автоматизованого обліку виконують низку важливих функцій, зокрема, фіксацію даних на електронних носіях, автоматичне документування, узагальнення й систематизацію даних для формування звітності, а також відображення результатів у друкованій або електронній формах. Вони забезпечують автоматизований контроль вихідних даних і оперативний аналіз інформації в процесі введення.

Вимоги до облікової інформації, що використовуються в автоматизованих системах, включають достовірність (точність та можливість перевірки даних), повноту (забезпечення максимуму корисних даних), своєчасність (отримання інформації у потрібний час), економічну доцільність (баланс між вигодами та витратами), порівнянність і постійність (здатність до аналізу динаміки) та обачність (обережне оцінювання активів і ризиків).

Головними перевагами таких систем є висока швидкість обробки даних, зниження ймовірності помилок, можливість отримання даних у реальному часі, гнучкість у формуванні звітності та ефективне використання ресурсів підприємства. Усе це дозволяє підприємствам досягати кращих результатів у бухгалтерському обліку та управлінні, мінімізуючи витрати часу і ресурсів.

Класифікація економічної інформації на підприємстві охоплює різноманітні аспекти, що визначають її характер, використання та передачу. За структурними підрозділами інформація може бути бухгалтерською, складською, виробничою, пов'язаною з реалізацією чи іншою. Залежно від періодичності вона поділяється на поточну (квартальна, річна) й оперативну (добова, місячна). Інформація стосовно підприємства як системи класифікується на вхідну, вихідну, внутрішню та зовнішню, а за стабільністю – на постійну, умовно-постійну та змінну. Ступінь обробки визначає поділ на первинну, проміжну та зведену інформацію. У відношенні до персоналу виділяють інформацію матеріально відповідальних осіб (комірника, касирів, завідувачів) та керуючу (директор, заступник, головний бухгалтер). Інформацію класифікують за об'єктом (якість товару, трудомісткість), формою передачі (вербальна, невербальна), стабільністю способу передачі, режимом передачі (примусова, за запитом), стадіями життєвого циклу товару (маркетинг, виробництво, продаж) і відношенням між об'єктом та суб'єктом управління (внутрішньо- та міжпідприємницька комунікація). Крім того, вона може бути достовірною чи недостовірною, корисною чи некорисною, плановою, нормативною, обліковою або аналітичною, активною чи пасивною, і поділяється на інформацію для матеріального та нематеріального виробництва, а також за галузями (харчова, хімічна тощо). Важливим є також поділ за стадіями управління (прогнозована, облікова) та за повнотою (надмірна, достатня). Застосування інформаційних технологій впливає на процеси документального забезпечення управління, який включає створення, передачу, реєстрацію, контроль та зберігання документів. Автоматизація дозволяє легко

формувати регламентовану та довільну звітність для зовнішніх і внутрішніх користувачів, забезпечуючи точність, оперативність і зручність. Це досягається завдяки використанню спеціалізованих програм, що підтримують формування стандартних, довільних та специфічних звітів, які відповідають міжнародним стандартам GAAP або IAS [1, с. 155].

## **1.2 Особливості побудови інформаційних систем бухгалтерського та управлінського обліку в вищих навчальних закладах**

Бухгалтерський облік – це невід'ємна складова системи управління будь-якої організації, включаючи вищі навчальні заклади (ВНЗ). Він являє собою упорядковану систему збору, реєстрації та узагальнення інформації про господарську діяльність у грошовому вираженні. Ця інформація охоплює активи, зобов'язання, капітал, доходи та витрати, і є основою для прийняття управлінських рішень, контролю за використанням ресурсів та складання фінансової звітності<sup>1</sup>.

Для отримання різнобічної інформації, необхідної для управління як окремим підприємством, так і економікою країни в цілому, використовують три взаємопов'язані види обліку: оперативно-технічний, бухгалтерський та статистичний<sup>2</sup>. Оперативно-технічний облік забезпечує поточну інформацію про хід господарської діяльності, бухгалтерський облік узагальнює інформацію про господарські операції в грошовому вираженні, а статистичний облік вивчає масові явища та процеси в економіці. Всі три види обліку взаємодоповнюють один одного та забезпечують керівництво необхідною інформацією для прийняття рішень.

Важливим аспектом бухгалтерського обліку є відображення змін у балансі під впливом господарських операцій. Бухгалтерський баланс – це звіт про фінансовий стан підприємства на певну дату, який відображає його активи, зобов'язання та капітал. Кожна господарська операція впливає на баланс,

змінюючи значення його статей<sup>1</sup>. Розуміння цих змін є ключовим для аналізу фінансового стану ВНЗ та прийняття обґрунтованих управлінських рішень.

Особливості обліку в бюджетних установах, до яких належать і ВНЗ, полягають у тому, що їх діяльність фінансується з державного бюджету. Це означає, що ВНЗ зобов'язані дотримуватися норм бюджетного законодавства та вести облік відповідно до встановлених правил<sup>3</sup>. Зокрема, вони повинні складати та виконувати кошторис доходів та видатків, а також звітувати про використання бюджетних коштів перед державними органами.

Нормативну базу бухгалтерського обліку в Україні складають:

Закон України «Про бухгалтерський облік та фінансову звітність в Україні»

Національні положення (стандарти) бухгалтерського обліку (П(С)БО), які регламентують методологію ведення обліку різних видів активів, зобов'язань, доходів та витрат<sup>6</sup>.

Інші нормативно-правові акти, такі як Положення про документальне забезпечення записів у бухгалтерському обліку, Положення про інвентаризацію активів та зобов'язань тощо.

Методичні рекомендації щодо облікової політики підприємства, які надають роз'яснення щодо застосування норм бухгалтерського обліку.

Міністерство фінансів України здійснює державне регулювання бухгалтерського обліку та визначає єдині методологічні засади його ведення<sup>8</sup>. Для бюджетних установ важливе значення мають також нормативні документи, що регулюють бюджетний процес та казначейське обслуговування.

Бухгалтерський облік в діяльності ВНЗ застосовується для:

- обліку надходження та використання бюджетних коштів;
- обліку доходів від надання платних послуг;
- обліку основних засобів, матеріалів та інших активів;
- нарахування та виплати заробітної плати;
- складання фінансової звітності.

Управлінський облік – це система збору, обробки та аналізу інформації, призначена для внутрішніх потреб управління підприємством. На відміну від бухгалтерського обліку, який орієнтований на зовнішніх користувачів (інвесторів, кредиторів, державні органи), управлінський облік надає інформацію для прийняття управлінських рішень керівництвом ВНЗ.

Основні функції управлінського обліку:

Планування: розробка бюджетів, прогнозування фінансових результатів, визначення цілей та завдань.

Контроль: моніторинг виконання планів, виявлення відхилень, аналіз причин відхилень.

Аналіз: оцінка ефективності діяльності, виявлення резервів, пошук шляхів покращення.

Прийняття рішень: надання інформації для обґрунтування управлінських рішень<sup>12</sup>.

Управлінський облік в ВНЗ може застосовуватися для:

- аналізу рентабельності освітніх програм;
- оцінки ефективності використання ресурсів;
- визначення оптимальної кількості студентів;
- формування цінової політики на платні послуги;
- контролю за витратами на наукові дослідження.

Важливим аспектом сучасних інформаційних систем обліку є автоматизація рутинних операцій. Наприклад, системи для закладів освіти дозволяють автоматизувати тарифікацію педагогічних працівників, розрахунок заробітної плати, облік батьківської плати, облік харчування тощо<sup>13</sup>. Це звільняє бухгалтерів від ручної праці та дозволяє їм зосередитися на аналізі даних та прийнятті рішень.

В українських ВНЗ використовуються різні інформаційні системи обліку, як вітчизняні, так і зарубіжні. Вибір системи залежить від потреб та можливостей конкретного ВНЗ.

Ключові відмінності в обліку бюджетних установ та комерційних підприємств полягають у наступному:

Джерела фінансування: бюджетні установи фінансуються з державного бюджету, комерційні підприємства – за рахунок власних коштів та доходів від діяльності.

Мета діяльності: бюджетні установи надають послуги населенню, комерційні підприємства – отримують прибуток.

Нормативне регулювання: бюджетні установи керуються бюджетним законодавством, комерційні підприємства – господарським.

Особливості інформаційних систем для бюджетних установ:

Ведення бюджетного обліку: облік надходження та використання бюджетних коштів, складання бюджетної звітності.

Казначейське обслуговування: інтеграція з системами казначейства для здійснення платежів та отримання фінансування.

Облік за кошторисом: контроль за виконанням кошторису доходів та видатків.

Важливою особливістю обліку в бюджетних установах є розмежування касових витрат та фактичних витрат. Касові витрати – це всі виплати, які здійснюються з рахунків установи, як готівкою, так і безготівково. Фактичні витрати – це постійні кінцеві витрати, оформлені документами, включаючи витрати по неоплачених рахунках та нараховану, але не виплачену заробітну плату<sup>12</sup>. Інформаційні системи для бюджетних установ дозволяють відстежувати та звіряти ці види витрат, що забезпечує достовірність обліку та звітності.

Приклади інформаційних систем для бюджетних установ:

"Парус-Бюджет": комплексна система для автоматизації обліку в бюджетних установах.

"IS-pro": система для управління фінансами та обліку в бюджетних організаціях.

### **1.3 Вимоги до сучасних інформаційних систем обліку в контексті цифрової трансформації**

Цифрова трансформація кардинально змінює підходи до ведення обліку та управління в усіх сферах, зокрема й у вищій освіті. Сучасні інформаційні системи обліку повинні відповідати низці вимог, щоб забезпечити ефективну підтримку діяльності ВНЗ в умовах цифрової економіки. Ці вимоги можна згрупувати за кількома ключовими аспектами:

#### **Функціональні вимоги:**

Повна автоматизація облікових процесів: Система повинна забезпечувати автоматизацію рутинних операцій, таких як введення первинних документів, формування проводок, складання звітності. Це дозволяє зменшити кількість помилок, звільнити час для аналітичної роботи та підвищити оперативність отримання інформації.

Підтримка різних видів обліку: Система повинна забезпечувати ведення як бухгалтерського, так і управлінського обліку, а також мати можливість інтеграції з іншими видами обліку (наприклад, кадровим, складським).

Гнучкість та масштабованість: Система повинна легко адаптуватися до змін в законодавстві, організаційній структурі ВНЗ та зростаючих обсягів даних.

Аналітичні можливості: Система повинна надавати інструменти для аналізу даних, формування звітів, візуалізації інформації та підтримки прийняття управлінських рішень. Важливими є можливості формування прогнозів, проведення факторного аналізу, побудови різноманітних звітів, як регламентованих, так і управлінських.

Інтеграція з іншими системами: Система повинна мати можливість інтеграції з іншими інформаційними системами, що використовуються у ВНЗ, такими як системи управління навчальним процесом (LMS), системи електронного документообігу, системи управління персоналом (HRM).

Підтримка електронного документообігу: Система повинна забезпечувати створення, обробку та зберігання електронних документів, що відповідають вимогам чинного законодавства.

Мобільність та доступність: Система повинна забезпечувати доступ до інформації з різних пристроїв (комп'ютери, планшети, смартфони) та з різних місць.

### **Технічні вимоги:**

Безпека даних: Система повинна забезпечувати захист даних від несанкціонованого доступу, втрати та пошкодження. Важливим є використання сучасних методів шифрування, аутентифікації та авторизації.

Надійність та продуктивність: Система повинна працювати стабільно та забезпечувати швидкий доступ до інформації.

Сучасна технологічна платформа: Система повинна бути розроблена на основі сучасних технологій, що забезпечують її масштабованість, гнучкість та можливість інтеграції з іншими системами.

Відповідність стандартам: Система повинна відповідати міжнародним та національним стандартам у сфері інформаційних технологій та обліку.

Вимоги до користувацького інтерфейсу:

Інтуїтивно зрозумілий інтерфейс: Система повинна мати простий та зручний інтерфейс, що не потребує спеціальних знань для використання.

Багатомовність: Для міжнародних ВНЗ важлива підтримка кількох мов.

Доступність для осіб з обмеженими можливостями: Система повинна відповідати вимогам доступності для осіб з обмеженими можливостями.

Вимоги щодо відповідності законодавству:

Система повинна відповідати чинному законодавству України у сфері бухгалтерського обліку, фінансової звітності та захисту персональних даних.

Врахування цих вимог дозволить створити або модернізувати інформаційну систему обліку, яка буде ефективно підтримувати діяльність ВНЗ в умовах цифрової трансформації та сприяти досягненню стратегічних цілей.

## 1.4 Огляд програмного забезпечення для автоматизації бухгалтерського та управлінського обліку

Комплекс ISpro — програмне забезпечення для автоматизації бухгалтерського, управлінського та оперативного обліку, орієнтоване на середні та великі підприємства, включаючи бюджетні та госпрозрахункові. Його функціонал охоплює аналіз бізнес-процесів і адаптується під різні профілі діяльності завдяки модульній архітектурі де кожна система відповідає за окремий аспект обліку або управління. Системи можуть функціонувати автономно або у складі комплексу.

Він забезпечує:

- Суцільний облік усіх або більшості аспектів діяльності підприємства.
- Безперервний облік із реєстрацією операцій у момент їх здійснення, розділений на облікові періоди.
- Взаємозалежний облік у єдиній базі даних, де введена інформація синхронно відображається у всіх необхідних ділянках.
- Документарний принцип зберігання даних, де кожен документ є базовою інформаційною одиницею.
- Облік у натуральних та вартісних вимірниках, включаючи багатовалютність і національну валюту як єдиний стандарт обліку.

До складу входять базові системи адміністрування та загальних довідників, які є необхідними для будь-якої конфігурації.

Комплекс включає наступні системи:

- Бухгалтерський та податковий облік у складі підсистем:
  - Загальна характеристика системи Головна книга
  - Загальна характеристика системи Книга покупок/продажів
- Керування фінансовими розрахунками у складі підсистем:
  - Загальна характеристика підсистеми Облік грошових коштів
  - Загальна характеристика підсистеми Ведення договорів

- Загальна характеристика підсистеми Розрахунки з контрагентами
- Облік основних засобів Загальна характеристика системи ;
- Логістика у складі підсистем:
- Загальна характеристика підсистеми Облік закупок
- Загальна характеристика системи Облік запасів
- Загальна характеристика системи Облік збуту
- Облік речового забезпечення Загальна характеристика системи ;
- Облік персоналу у складі підсистем:
- Загальна характеристика системи Облік кадрів
- оху\_ex-1/epr/epr.ditamap Електронна прохідна
- Загальна характеристика системи Облік праці та заробітної плати
- Загальна характеристика системи Менеджмент персоналу
- Загальна характеристика системи Управління студентами
- Загальна характеристика системи Облік автотранспорту
- Загальна характеристика системи Облік виробництва
- Загальна характеристика системи Фінансове планування й аналіз
- Загальна характеристика системи Документообіг та управління

процесами

- Загальна характеристика системи Відносини з клієнтами
- Загальна характеристика системи Облік комунальних послуг
- Загальна характеристика системи Загальні довідники
- Загальна характеристика системи Сервіс;
- Загальна характеристика системи Адміністратор[5, с 5]

Набір систем налаштовується відповідно до потреб підприємства: для виробничих компаній доступний модуль управління виробництвом, який не є обов'язковим для бюджетних організацій. Крім того, за потреби можуть додаватися спеціалізовані функції, як-от облік за джерелами фінансування, що закуповуються окремо.

ISpro дозволяє використовувати мобільні додатки, які є додатковими інструментами:

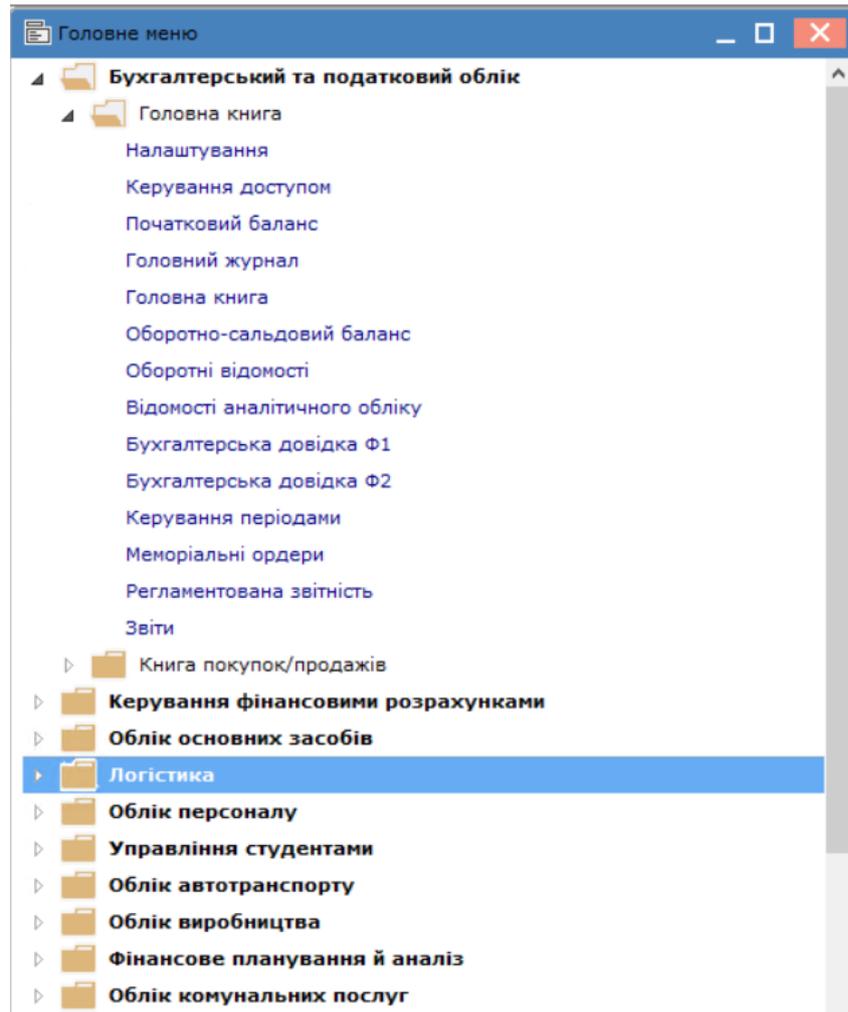
- ISpro: Profile ;
- ISpro: Link ;
- ISpro: Tasks ;
- ISpro: BPM ;
- ISpro: InStock .
- ISpro: Workflows

Бухгалтерський облік у Комплексі ISpro може вестися паралельно з іншими видами обліку або відкладено в часі, забезпечуючи проведення первинних господарських документів та створення масиву проводок, які заносяться до Головного журналу та автоматично формують Головну книгу, Журнали-ордери, оборотно-сальдовий баланс і всі необхідні звіти. Податковий облік може також вестися паралельно з бухгалтерським або відкладено, зі створенням окремого плану рахунків. Управлінський облік включає сервісні можливості для аналізу діяльності та прийняття управлінських рішень, можливе також створення окремого плану рахунків. Оперативний облік базується на первинних документах і не вимагає негайного формування проводок, дозволяючи швидко знаходити та виправляти помилки у документах та коригувати проводки перед закінченням звітного періоду, враховуючи останні зміни законодавства.

## **1.5 Переваги та недоліки використання ISpro для автоматизації обліку**

### **Переваги:**

1. Універсальність: ISpro надає комплексне рішення для управління всіма аспектами діяльності підприємства, включаючи бухгалтерський, податковий, кадровий та оперативний облік.



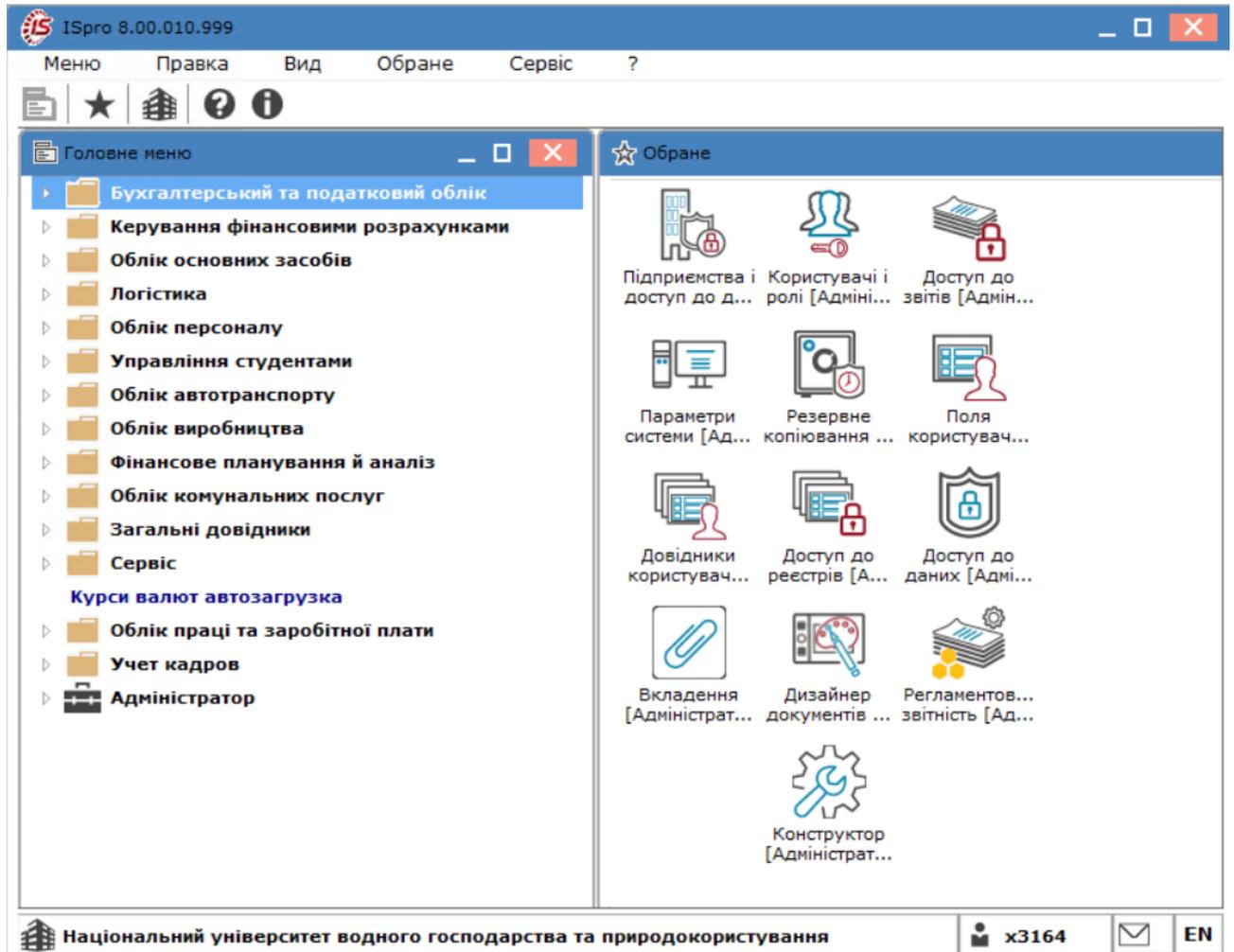


Рис 1.1. Головне меню

Комплекс має ієрархічну структуру підсистем і модулів. Вікно Головного меню можна представити із виділенням обраних модулів. Праворуч від Головного меню знаходиться вікно Обране, куди за допомогою миші або по клавіші F5 переносяться модулі, що необхідні для користування. Таке представлення при роботі спрощує пошук необхідного модуля у підсистемах. Видалення з обраного непотрібного модуля проводиться по клавіші F8 або delete.

2. Автоматизація: Система дозволяє автоматизувати рутинні завдання, звільняючи час для стратегічних завдань. Централізований розрахунок значно прискорює розрахунок заробітної плати при великій кількості працівників на підприємстві. Дані розрахунку представлені у вигляді реєстру виконаних розрахунків, де відображається Код та Найменування,

Чисельність, Період розрахунку, кількість Розраховано, Дата, Час, Хто виконав, Телефон, Пріоритет розрахунку.

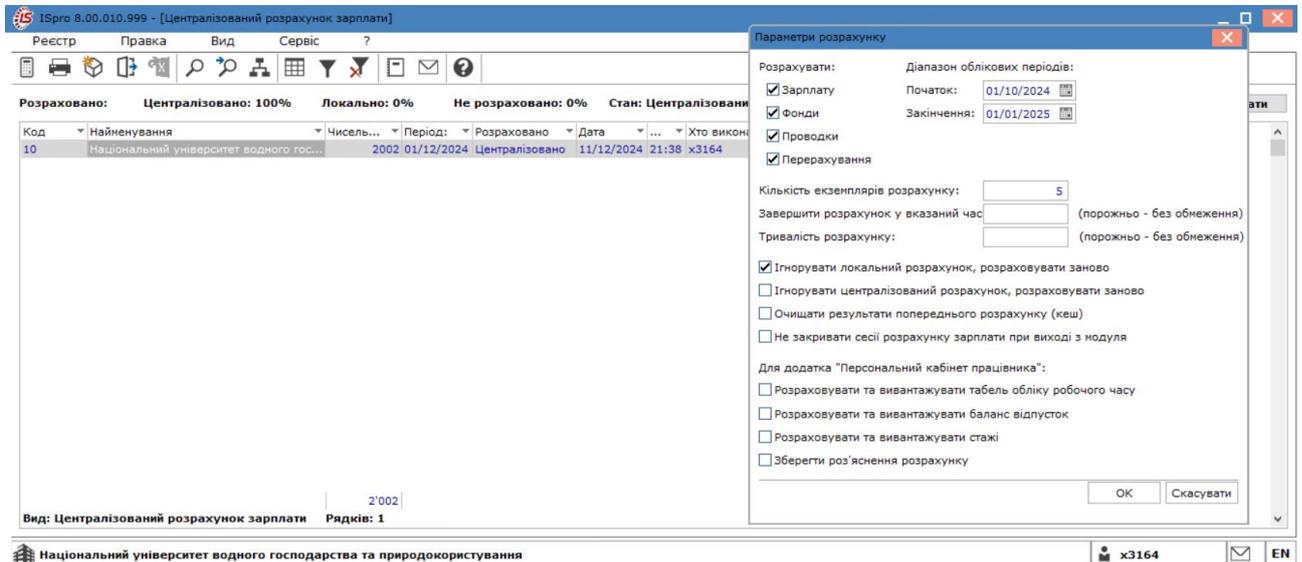


Рис 1.2. Параметри налаштування проведення централізованого розрахунку зарплати

Активувавши комбінацію клавіш Alt+F5 запуститься розрахунок зарплати. У разі необхідності, зупинка проведення розрахунку виконується по пункту меню Реєстр/Зупинити розрахунок (комбінація клавіш Ctrl+F5).

- Гнучкість налаштувань: ISpro дозволяє налаштовувати облікові процеси відповідно до специфіки підприємства та змін у законодавстві, що забезпечує високу адаптивність системи. Обмін даними між різними системами (веб-системою, десктопною версією Комплексу, мобільними додатками) у REST API здійснюється за протоколом HTTP.

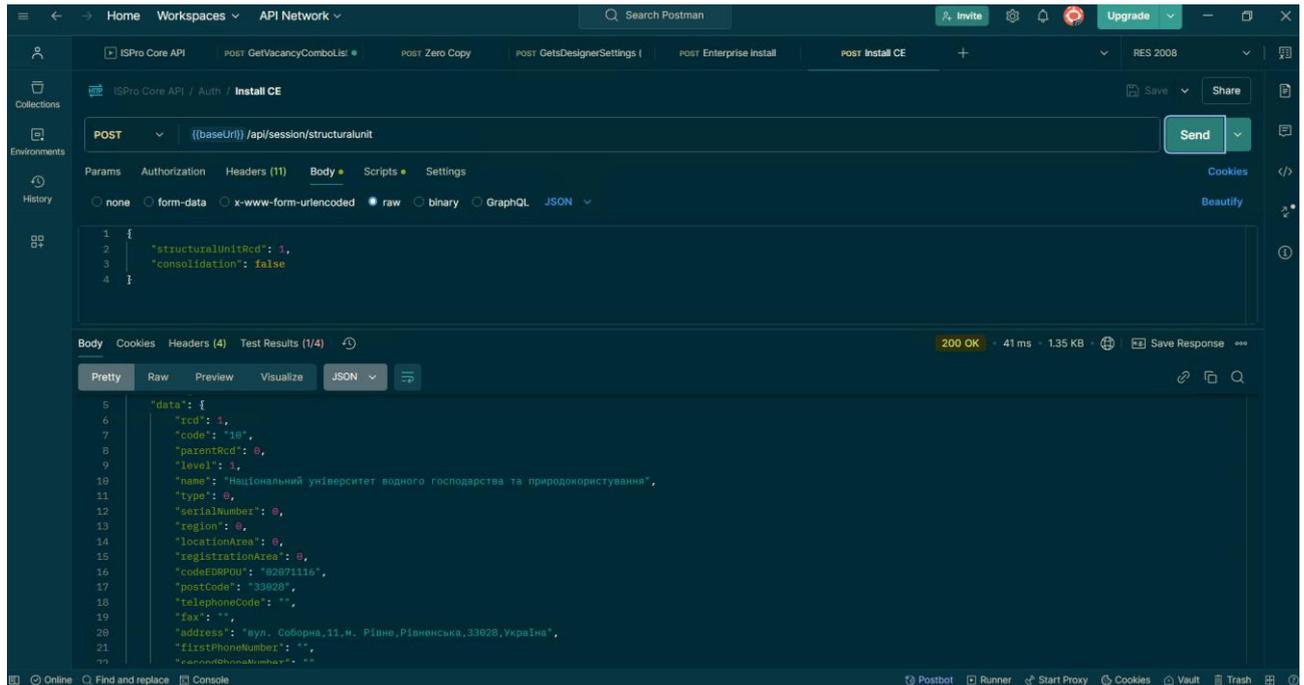


Рис 1.3 Робота з сесією користувача

У свою чергу, REST API ISpro надає список URL-адрес, за допомогою яких сервер аплікацій може приймати запити на отримання, збереження, оновлення і видалення даних.

4. **Захист і безпека даних:** ISpro забезпечує захист даних від несанкціонованого доступу та втрати, що є критично важливим для бухгалтерської інформації Додаток А. Систему підключено до платформи MISIP-UA, ситуаційного центру забезпечення кібербезпеки Служби безпеки України. Платформа широко використовується в усьому світі, відповідає міжнародним стандартам ЄС та НАТО, застосовується основними міжнародними суб'єктами у сфері кібербезпеки FIRST, CIRCL, CiviCERT, NATO NCI Agency.
5. **Контроль:** Система забезпечує повний контроль над фінансовими потоками, кадровими змінами та операціями.
6. **Звітність:** ISpro генерує широкий спектр звітів, допомагаючи в прийнятті обґрунтованих рішень.

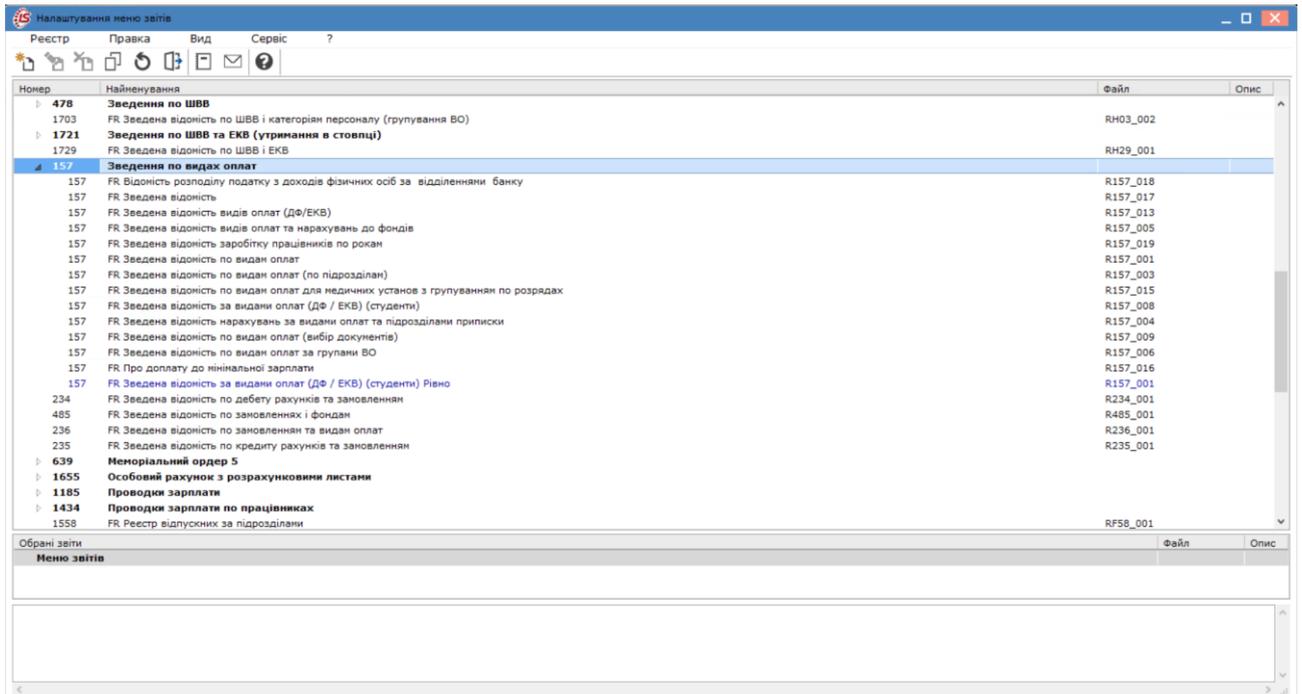


Рис 1.5. Меню звітів

Серед широкого списку звітів можна обрати потрібні натиснувши на крайню з ліва кнопку. У будь-якому вікні реєстру звітів крім шаблонів друку є кнопка Експорт поточного представлення, при натисканні якої, здійснюється експорт даних в Excel. При відсутності програми Excel на комп'ютері користувача і наявності програми OpenOffice Calc, реєстр звітів буде відкриватися в OpenOffice Calc.

7. Підтримка: Кваліфікована служба технічної підтримки готова допомогти вирішити будь-які питання.

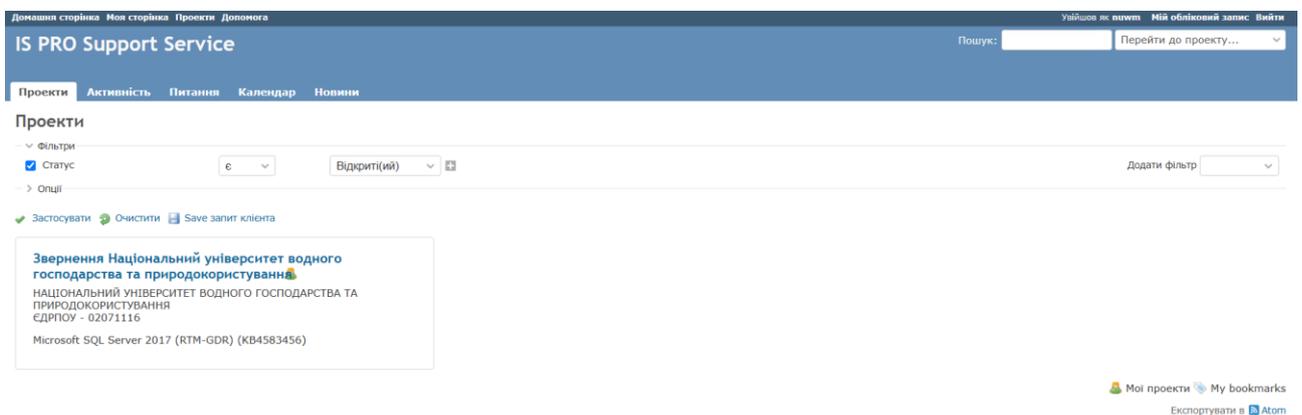


Рис 1.6 HelpDesk ISpro

Платформа призначена для ефективного управління зверненнями та обслуговування клієнтів. Вона дозволяє автоматизувати всі процеси обробки запитів, створюючи централізовану систему управління. Завдяки цій платформі можна оперативно приймати рішення, відстежувати статус кожного звернення та оптимізувати операційні витрати.

## Недоліки:

1. **Складність впровадження:** Необхідність глибоких знань та навичок для ефективного впровадження та налаштування системи, як приклад можна взяти дизайнер звітів.

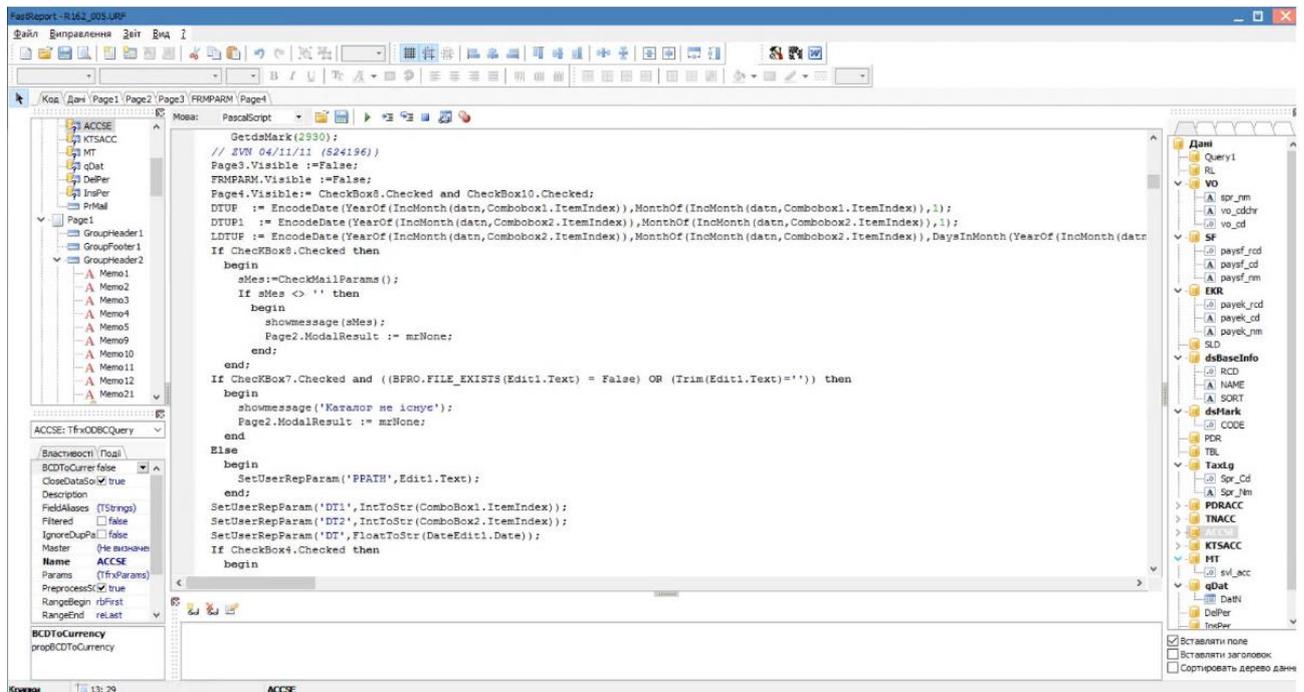


Рис 1.7. Дизайнер FastReport

Необхідність створення форм, обробників подій і складної логіки звітів вимагає від користувача значного технічного досвіду. Це ускладнює впровадження системи в організаціях без кваліфікованих фахівців, затримуючи процес і підвищуючи витрати на навчання або залучення зовнішніх експертів.

**Залежність від постачальника:** Залежність від одного постачальника програмного забезпечення створює ризики, пов'язані з обмеженою підтримкою оскільки проблеми роботи програми пов'язані з базою даних у кожному

підприємстві відрізняються, невелика кількість спеціалістів не дивлячись на те що реалізована платформа для спрощення комунікації вирішення значної кількості проблем потребує чимало часу.

Нове завдання #276676 [ВІДКРИТЕ](#) ✎ Редагувати ★ Дивитися ...

**Переоцінка валюти**

Доданий buhgalteria nuwm близько 1 місяця тому. Оновлений 8 днів тому.

« Попередній | 20/44 | Наступний »

<b>Статус:</b>	Передано замовнику	<b>Початок:</b>	04.11.2024
<b>Пріоритет:</b>	Високий	<b>Дата виконання:</b>	
<b>Призначена до:</b>	Кот Тетяна	<b>% зроблено:</b>	<div style="width: 100px; height: 10px; background-color: #ccc; border: 1px solid #ccc;"></div> 0%
<b>Siid:</b>	<a href="https://siid.ispro.ua/issues/169340">https://siid.ispro.ua/issues/169340</a>	<b>Оцінний час:</b>	
		<b>Модуль:</b>	Облік грошових коштів

**Опис** 💬 Цитувати

Не вірно відображено переоцінку валюти у виписці банку, а саме долар за 10.10.2024 у валюті долар. У виписці у валюті гривня переоцінка вірна.  
Старікова І.В.

**Файли**

[Без імені1.png](#) (139 КБ) buhgalteria nuwm, 04.11.2024 11:33

[Без імені.png](#) (138 КБ) buhgalteria nuwm, 04.11.2024 11:33

**Підзадачі** ➕ Додати

Історія | **Notes** | Property changes

Оновлено Кот Тетяна близько 1 місяця тому 💬 ... #3

- Параметр Статус змінився з У роботі на Проблема
- Параметр Siid змінився на <https://siid.ispro.ua/issues/169340>

Рис 1.8. Завдання розміщене на платформі helpdesk.ispro.ua

Завдання, пов'язане з переоцінкою валюти, демонструє, як проблеми можуть виникати через специфічні налаштування бази даних та скільки часу може займати їх вирішення.

Завдяки інтегрованості, гнучкості налаштувань та підтримці прийняття рішень, ISpro стає незамінним інструментом для багатьох підприємств. Однак, як і будь-яке програмне забезпечення, вона має свої обмеження, серед яких складність впровадження та залежність від постачальника. Важливо враховувати ці аспекти, обираючи ISpro для автоматизації облікових процесів, адже її ефективність значною мірою залежить від правильного налаштування та підтримки.

## **2 РОЗДІЛ. Аналіз стану інформаційної системи бухгалтерського та управлінського обліку НУВГП**

### **2.1 Загальна характеристика НУВГП та його інформаційної системи**

Національний університет водного господарства та природокористування (НУВГП) – заклад вищої освіти IV р.а. Університет є навчально-науковим комплексом, що здійснює підготовку висококваліфікованих фахівців, науково-педагогічних кадрів, забезпечує підвищення кваліфікації фахівців та проводить науково-дослідну роботу. До структури університету станом на 31.12.2023 входять 8 навчально-наукових інститутів: енергетики, автоматики та водного господарства (ННІЕАВГ); механічний (ННМІ); агроєкології та землеустрою (ННІАЗ); будівництва та архітектури (ННІБА); кібернетики, інформаційних технологій та інженерії (ННІКІТІ); економіки та менеджменту (ННІЕМ); права (ННІП); охорони здоров'я (ННІОЗ) та інститут післядипломної освіти (ІПО), які загалом об'єднують 49 кафедр. В університеті як окремі підрозділи працюють кафедри військової підготовки та фізичного виховання.

Університет здійснює освітню діяльність відповідно до відомостей про право здійснення освітньої діяльності закладами вищої освіти, розміщених на сайті МОН України, за наступними рівнями вищої освіти: фахова передвища освіта за 26 спеціальностями; початковий рівень (короткий цикл) вищої освіти за 3 спеціальностями; перший (бакалаврський) рівень вищої освіти за 50 спеціальностями; другий (магістерський) рівень вищої освіти за 41 спеціальністю; третій (освітньо-науковий) рівень вищої освіти за 12 спеціальностями[8].

Ведення бухгалтерського обліку і фінансової звітності здійснюється відповідно до:

наказу МФУ від 31.12.2013 № 1203 «Про затвердження Плану рахунків бухгалтерського обліку бюджетних установ» із змінами та доповненнями;

наказу Міністерства фінансів України від 12.03.2012 № 333 «Про затвердження Інструкції щодо застосування економічної класифікації видатків бюджету» із змінами;

наказу МФУ від 12.10.2010 № 1202 «Про затвердження національних положень (стандартів) бухгалтерського обліку в державному секторі» із змінами та доповненнями;

наказу МФУ 08.09.2017 № 755 «Про затвердження типових форм меморіальних ордерів, інших облікових реєстрів суб'єктів державного сектору та порядку їх складання»;

наказу МФУ 29.06.2017 № 604 «Про затвердження форм карток і книг аналітичного обліку суб'єктів державного сектору та порядку їх складання»;

наказу МФУ від 29.12.2015 № 1219 «Про затвердження деяких нормативно-правових актів з бухгалтерського обліку в державному секторі» із змінами;

наказу МФУ від 23.01.2015 № 11 «Про затвердження Методичних рекомендацій з бухгалтерського обліку для суб'єктів державного сектору» із змінами;

постанови КМУ від 08.11.2007 № 1314 «Про затвердження Порядку списання об'єктів державної власності» із змінами; наказ МФУ від 13.09.2016 № 818 «Про затвердження типових форм з обліку та списання основних засобів суб'єктами державного сектору та порядку їх складання»;

наказу МФУ від 02.04.2014 № 372 «Про затвердження Порядку бухгалтерського обліку окремих активів та зобов'язань бюджетних установ та внесення змін до деяких нормативно-правових актів з бухгалтерського обліку бюджетних установ» із змінами;

наказу МФУ від 02.09.2014 № 879 «Про затвердження Положення про інвентаризацію активів та зобов'язань» із змінами; наказу МФУ від 17.06.2015 № 572 «Про затвердження типових форм для відображення бюджетними установами результатів інвентаризації»;

Податкового кодексу України зі змінами;

Бюджетного кодексу України зі змінами;  
національних положень (стандартів) бухгалтерського обліку (НП(С)БО);  
інших нормативних документів.

Закону України № 996-ХІУ від 16.07.1999 р. “Про бухгалтерський облік та фінансову звітність в Україні”, який визначає правові засади регулювання, організації, ведення бухгалтерського обліку та складання фінансової звітності в Україні.

До структури бухгалтерії входять:

Заступники головної бухгалтерки

Бухгалтерія по обліку матеріальних цінностей та податків

Бухгалтерія по стипендії

Бухгалтерія по гуртожитках, розрахунках по банку

Бухгалтерія по зарплаті

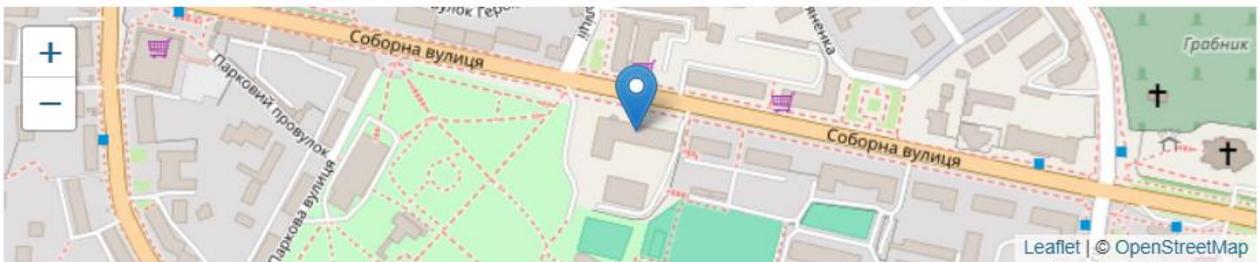
Бухгалтерія по платному навчанні

Каса

На цей час в бухгалтерії працюють 16 спеціалістів очолює Матвієнко Марія Дмитрівна. Вона підпорядковується безпосередньо Ректору.

## БУХГАЛТЕРІЯ. КАСА

- ☎ Головна бухгалтерка - вн. 6-28; 0 (362) 63-32-18
- ☎ Заступниця головної бухгалтерки: вн. 6-25; 0 (362) 63-34-41
- ☎ Бухгалтерія по обліку матеріальних цінностей та податків: вн. 7-36; 0 (362) 63-46-69
- ☎ Бухгалтерія по стипендії: вн. 6-06; (0362) 63-34-41
- ☎ Бухгалтерія по гуртожитках, розрахунках по банку: вн. 7-07; 0 (362) 63-46-69
- ☎ Бухгалтерія по зарплаті: вн. 7-34; 0 (362) 63-41-13
- ☎ Бухгалтерія по платному навчанні: вн. 6-88; 0 (362) 63-59-60
- ☎ Каса: вн. 7-06; 0 (362) 26-95-41
- 📍 Адреса: м. Рівне, вул. Соборна 11, навчальний корпус №1, каб. 153-158
- ✉ [buhgalteria@nuwm.edu.ua](mailto:buhgalteria@nuwm.edu.ua)



Діяльність

Склад відділу

Бухгалтерський облік в університеті ведеться згідно Закону України № 996-ХІУ від 16.07.1999 р. "Про бухгалтерський облік та фінансову звітність в Україні", який визначає правові засади регулювання, організації, ведення бухгалтерського обліку та складання фінансової звітності в Україні.

Рис.2.1 Сторінка Бухгалтерії на сайті НУВГП

Загальний бюджет університету відображені в таблиці 2.1 за 2023 рік.

Таблиця 2.1

## Надходження грошових коштів за загальним фондом бюджету 2023 рік

Найменування показника	Сума коштів, тис. грн
1	2
<b>2201160, у тому числі:</b>	<b>152 665,38</b>
- заробітна плата та нарахування	138 467,36
- придбання матеріалів, одягу для дітей-сиріт	224,98
- харчування дітей-сиріт	2 176,09
- комунальні та інші послуги	11 301,56
- грошова матеріальна допомога дітям-сиротам	495,39
<b>2201040 прикладні розробки, у тому числі:</b>	<b>4 781,89</b>
- заробітна плата та нарахування	4 221,91
- придбання матеріалів	363,27
- відрядження	79,35
- комунальні та інші послуги	117,36
<b>2201190 академічні стипендії студентам, аспірантам, докторантам</b>	<b>28 542,59</b>
<b>2201080 державні премії, стипендії та гранти в галузі освіти, науки і техніки, стипендії переможцям міжнародних конкурсів</b>	<b>322,08</b>

У 2023 році було отримано 136 з половиною мільйонів гривень як плату за послуги.

## 2.2 Аналіз бізнес-процесів бухгалтерського та управлінського обліку в НУВГП

Перед впровадженням інформаційної системи обліку був проведений аналіз бізнес-процесів бухгалтерського та управлінського обліку в НУВГП. Він складався з наступних етапів.

### 1 Збір інформації

Документація процесів:

Зібрано регламенти, положення та посадові інструкції, пов'язані з бухгалтерським та управлінським обліком.

Вивчено законодавчі вимоги, що стосуються обліку в державних установах.

Фітбек з ключовими співробітниками:

Проведено інтерв'ю з головним бухгалтером, економістами та працівниками управління фінансів.

Виявлено ключові проблеми, зокрема дублювання ключового поля при одночасній роботі кількох користувачів у одній таблиці та тривалий час обробки документів.

Аналіз документообігу та інформаційних потоків:

Виявлено, що значна частина інформації обробляється вручну, що призводить до помилок.

Більшість документів зберігаються у паперовому вигляді, що уповільнює доступ до даних та збільшує витрати на забезпечення друку.

Використання програмного забезпечення:

Університет використовує застарілу версію бухгалтерської програми 1С, яка не інтегрована з іншими системами.

Відсутня єдина інформаційна платформа для управлінського обліку.

## **2 Мапування процесів**

### **1. Підготовка фінансових звітів**



Рис. 2.2 Процес підготовки фінансових звітів

Процес підготовки фінансових звітів починається на початку місяця зі збору даних з різних відділів. Після перевірки даних на повноту та точність, вони консолідуються, розраховуються необхідні показники та формується звіт. Процес завершується поданням готового фінансового звіту.

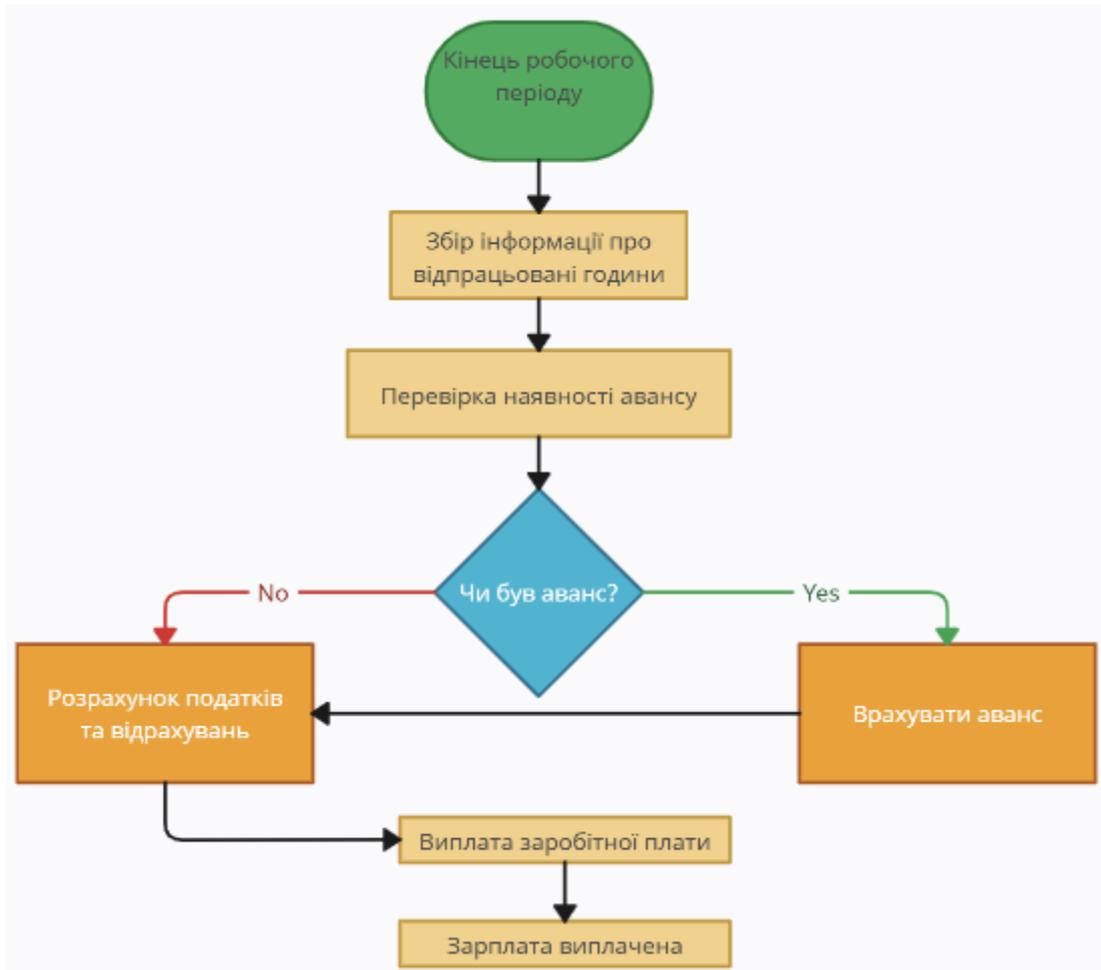


Рис. 2.3 Процес розрахунку заробітної

Процес розрахунку заробітної плати починається з кінця робочого періоду, коли збирається інформація про відпрацьовані години. Після цього перевіряється наявність авансу, і якщо він був, враховується при розрахунку. Далі здійснюється розрахунок податків та відрахувань, після чого виплачується заробітна плата, завершуючи процес.

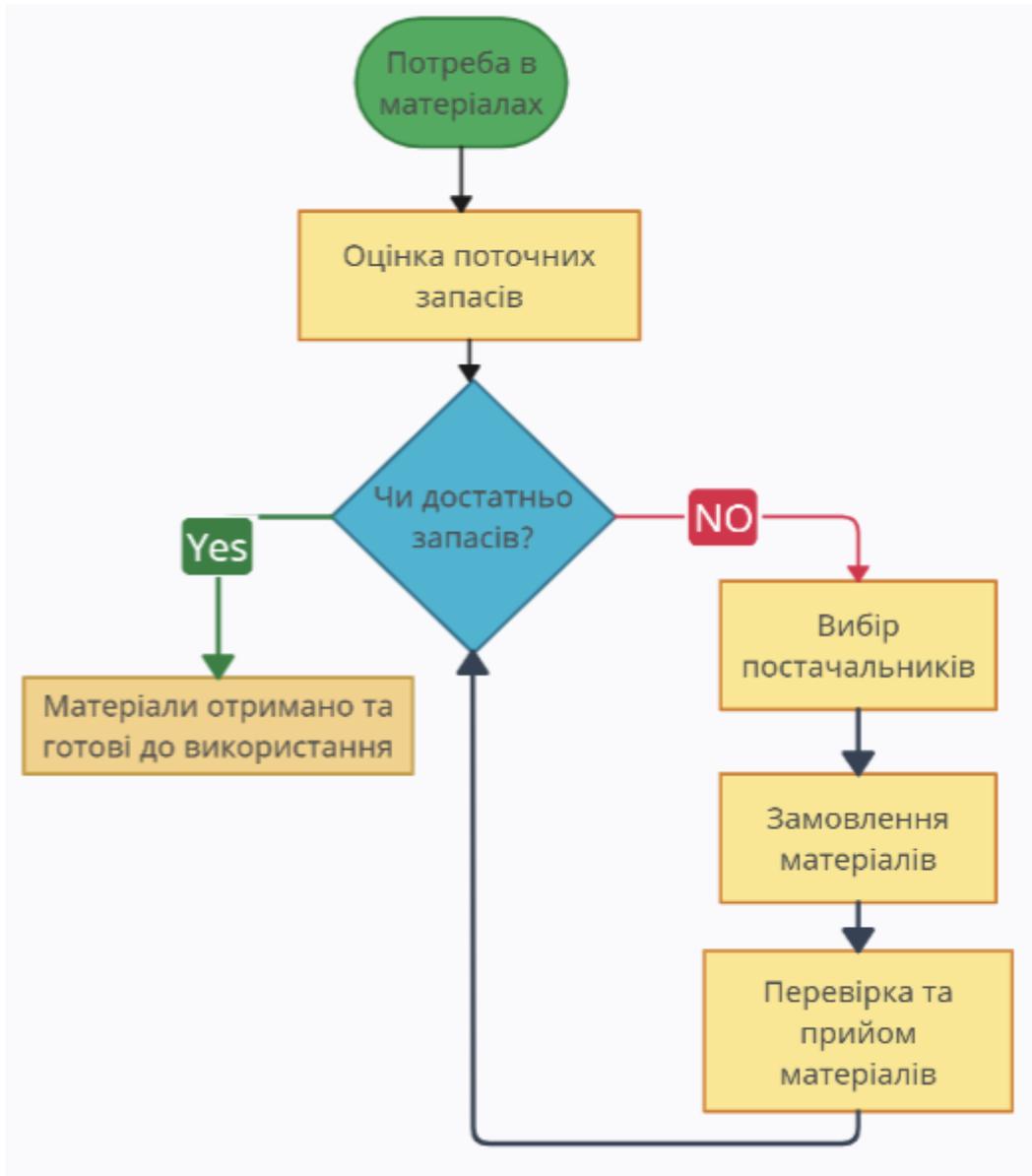


Рис.2.4 Процес управління витратами

Процес управління витратами на матеріально-технічне забезпечення починається з виявлення потреби в матеріалах. Після оцінки поточних запасів приймається рішення про необхідність замовлення. Якщо запаси недостатні, вибираються постачальники, замовляються матеріали, які потім перевіряються та приймаються. Процес завершується, коли матеріали отримано та готові до використання.

Проведений аналіз процесів бухгалтерського та управлінського обліку в НУВГП виявив низку проблем та можливостей для вдосконалення. Документація процесів показала, що значна частина інформації обробляється

вручну, що призводить до помилок та затримок. Використання застарілої версії бухгалтерської програми 1С, яка не інтегрована з іншими системами, також ускладнює роботу.

Інтерв'ю з ключовими співробітниками виявили проблеми дублювання ключового поля та тривалий час обробки документів. Аналіз документообігу та інформаційних потоків підтвердив, що більшість документів зберігаються у паперовому вигляді, що уповільнює доступ до даних та збільшує витрати на забезпечення друку.

Мапування процесів підготовки фінансових звітів, розрахунку заробітної плати та управління витратами на матеріально-технічне забезпечення дозволило виявити ключові етапи та можливі точки оптимізації. Впровадження сучасних інформаційних систем, таких як ISpro, дозволить автоматизувати процеси, зменшити кількість помилок та підвищити ефективність роботи.

Загалом, вдосконалення інформаційної системи бухгалтерського та управлінського обліку на основі ISpro сприятиме підвищенню точності, прозорості та ефективності облікових процесів в університеті.

### **2.3 Аналіз ефективності функціонування ISpro**

#### Основні компоненти ISpro

ISpro - це модульна ERP-система, що дозволяє налаштовувати її під потреби конкретного підприємства. Основні компоненти системи включають:

**База даних:** Серце системи, де зберігаються всі дані про підприємство: фінанси, облік, персоналу, виробництво тощо.

**Модулі:** Функціональні блоки системи, що відповідають за різні аспекти управління підприємством (фінанси, облік, виробництво, логістика тощо).

**Інтерфейс користувача:** Забезпечує взаємодію користувача з системою, дозволяє вводити, редагувати та аналізувати дані.

**Сервер додатків:** Запускає та обробляє запити користувачів до системи.

Інтеграційні інтерфейси: Дозволяють підключати ISpro до інших систем і сервісів.

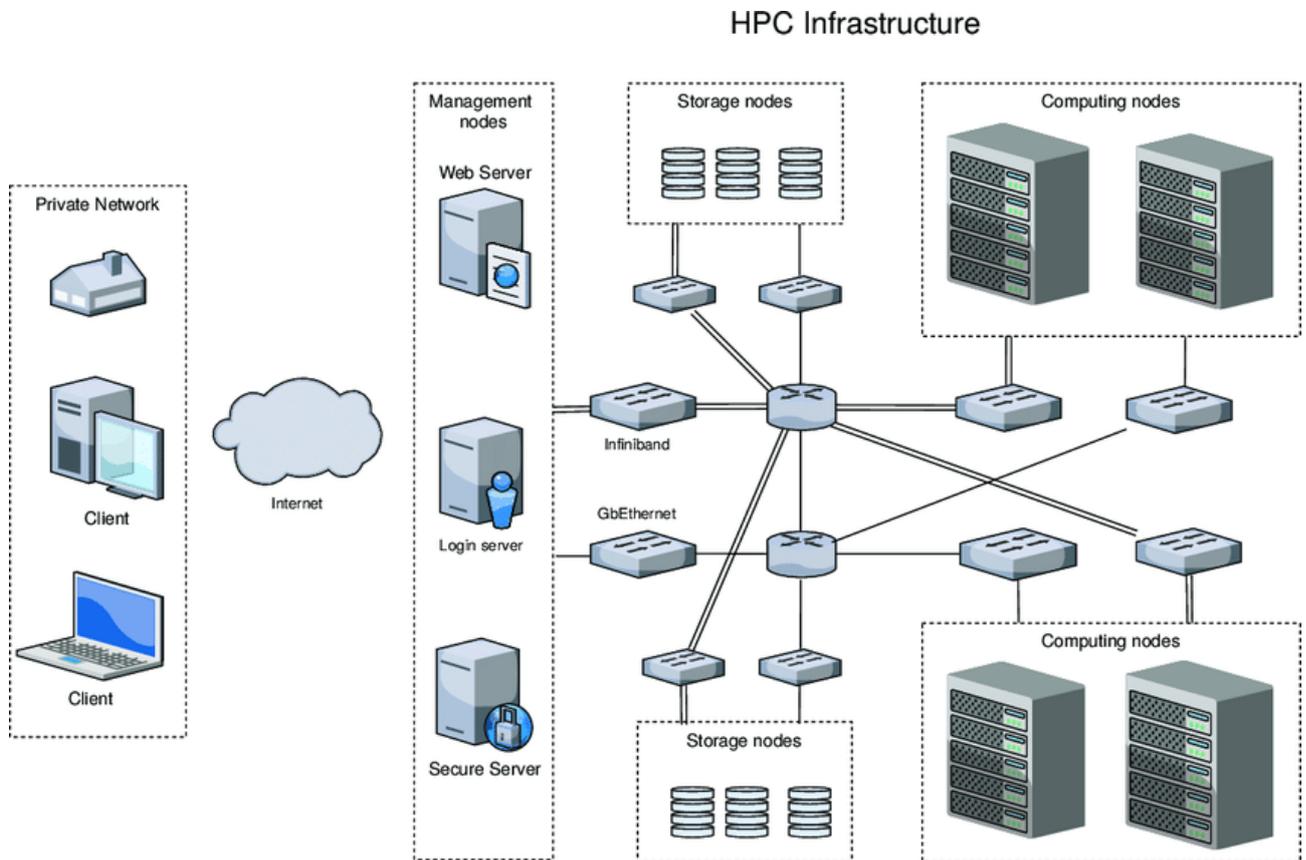


Рис 2.5 Спрощений приклад архітектури системи високопродуктивних обчислень

### Типова структура ISpro включає наступні рівні:

Рівень даних: База даних, де зберігаються всі дані підприємства.

Рівень бізнес-логіки: Модулі, що реалізують бізнес-процеси підприємства.

Рівень користувальницького інтерфейсу: Веб-інтерфейс, через який користувачі взаємодіють з системою.

### Взаємозв'язки між компонентами

База даних: Зберігає всі дані, які використовуються в системі. Модулі отримують дані з бази даних і записують в неї результати своєї роботи.

Модулі: Взаємодіють між собою через базу даних. Наприклад, модуль "Продажі" може передавати дані про замовлення модулю "Виробництво".

Інтерфейс користувача: Забезпечує доступ користувача до даних і функціоналу системи. Користувач через інтерфейс може вводити дані, запускати процеси, отримувати звіти.

Сервер додатків: Обробляє запити користувачів, запускає модулі, забезпечує взаємодію між компонентами системи.

Інтеграційні інтерфейси: Дозволяють підключати ISpro до інших систем, таких як бухгалтерські системи, системи електронного документообігу тощо.

Таблиця 2.2

#### Технічні характеристики

Компонент	Деталі
База даних	Microsoft SQL Server 2019 Enterprise Edition
Сервер додатків	Microsoft.NET Framework 4.0
Веб-сервер	Microsoft IIS 10.0
Операційна система	Windows Server 2016
Мережа	Локальна мережа Гбіт/с

**Інтеграція ISpro** з іншими інформаційними системами є ключовим аспектом успішного впровадження та використання цієї ERP-системи. Вона дозволяє об'єднати різноманітні дані та процеси, забезпечити безперервність бізнес-процесів та підвищити ефективність роботи організації.

#### Способи інтеграції

Файловий обмін: Дані експортуються в певний формат файлу (наприклад, CSV, XML) і імпортуються в іншу систему.

Веб-сервіси: Використання веб-сервісів (SOAP, REST) для обміну даними в реальному часі.

Бази даних: Створення прямого з'єднання між базами даних різних систем.

Інтеграційні платформи: Використання спеціалізованих інтеграційних платформ (наприклад, MuleSoft, Dell Boomi) для створення єдиної інтеграційної шини.

### **Переваги інтеграції ISpro**

Збільшення ефективності: Автоматизація ручних процесів, зменшення кількості помилок.

Покращення якості даних: Забезпечення єдиного джерела правдивої інформації.

Підвищення прозорості бізнес-процесів: Створення єдиного інформаційного простору.

Зменшення витрат: Оптимізація використання ресурсів.

### **Типи інтеграції ISpro**

Інтеграція з іншими ERP-системами: Обмін даними між різними ERP-системами, що використовуються в організації (наприклад, при злитті компаній або зміні основної ERP-системи).

Інтеграція з бухгалтерськими системами: Синхронізація фінансових даних, таких як рахунки, платежі, дебіторська та кредиторська заборгованість.

Інтеграція з системами управління персоналом (HRM): Обмін даними про співробітників, розрахунок заробітної плати, відпустки тощо.

Інтеграція з системами електронного документообігу (ECM): Автоматизація процесів обробки документів, їх зберігання та архівування.

Інтеграція з системами управління запасами: Синхронізація даних про наявність товарів на складах, замовлення та поставки.

Інтеграція з веб-сайтами та інтернет-магазинами: Обмін інформацією про продукти, ціни, наявність на складі.

Інтеграція з банківськими системами: Автоматизація процесів платежів, звірки банківських виписок.

Інтеграція з державними реєстрами: Обмін даними з державними реєстрами (наприклад, податкова, пенсійний фонд).

## Виклики при інтеграції

Сумісність систем: Різні системи можуть використовувати різні формати даних, протоколи та технології.

Складність процесів: Інтеграція може бути складним процесом, який вимагає знань і досвіду.

Період: 15/12/2024

№ картки	Табельни...	Прізвище Ім'я По батькові	Підрозділ	Посада	Дата при...	Дата з...
100182	100182	Грицина Олександр Олександрович	Навчально-науковий Північний локаль...	Доцент	08/09/2016	30/06/...
100994	100994	Шевченко Катерина Павлівна	Навчально-науковий Північний локаль...	Провідний фахівець	04/01/2010	01/11/...
215	215	Вечер В'ячеслав Валентинович	Навчально-науковий Північний локаль...	Доцент	02/09/2020	30/06/...
216	216	Сонечко Світлана Аркадіївна	Навчально-науковий Північний локаль...	Бібліотекар I категорії	15/09/2020	28/06/...
481	481	Клюха Оксана Олександрівна	Навчально-науковий Північний локаль...	Доцент	23/11/2020	30/06/...
583	583	Павелко Ольга Віталіївна	Навчально-науковий Північний локаль...	Професор	22/10/2020	28/06/...
584	584	Дорошенко Олена Олександрівна	Навчально-науковий Північний локаль...	Доцент	22/10/2020	30/06/...
585	585	Рудик Андрій Вікторович	Навчально-науковий Північний локаль...	Професор	26/11/2020	30/06/...
586	586	Цесик Ярослав Петрович	Навчально-науковий Північний локаль...	Доцент	23/10/2020	30/06/...
590	590	Коротун Серій Ігорович	Навчально-науковий Північний локаль...	Доцент	23/10/2020	30/06/...
591	591	Вечер В'ячеслав Валентинович	Навчально-науковий Північний локаль...	Доцент	23/10/2020	30/06/...
592	592	Янчук Олександр Євгенович	Навчально-науковий Північний локаль...	Доцент	23/10/2020	30/06/...
595	595	Грицина Олександр Олександрович	Навчально-науковий Північний локаль...	Доцент	23/10/2020	30/06/...
824	824	Пікула Нікола Володимирович	Навчально-науковий Північний локаль...	Старший викладач	06/11/2020	30/06/...
1029	1029	Котеленець Любов Миколаївна	Навчально-науковий Північний локаль...	Прибиральник службових приміщень	01/02/2021	26/02/...
894	894	Скаковська Світлана Степанівна	Навчально-науковий Північний локаль...	Доцент	27/10/2020	30/06/...
915	915	Чернікова Ольга Петрівна	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...
917	917	Лішинський Анатолій Георгійович	Навчально-науковий Північний локаль...	Доцент	26/10/2020	30/06/...
920	920	Легшина Олена Леонідівна	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...
919	919	Погорелова Валентина Миколаївна	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...
922	922	Маркович Марія Миколаївна	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...
925	925	Кінчур Олександр Федорович	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...
928	928	Анцеева Лідія Володимирівна	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...
930	930	Завець Лариса Борисівна	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...
932	932	Ершова Лариса Василівна	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...
934	934	Кочубей Алла Володимирівна	Навчально-науковий Північний локаль...	Доцент	26/10/2020	30/06/...
937	937	Лашенко Галина Віталіївна	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...
962	962	Нещадко Анатолій Анатолійович	Навчально-науковий Північний локаль...	Доцент	26/10/2020	30/06/...
963	963	Заріпов Артем Рахімович	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...
965	965	Суддя Володимир Андрійович	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...
966	966	Волк Павло Павлович	Навчально-науковий Північний локаль...	Доцент	26/10/2020	30/06/...
967	967	Пічук Олег Леонідович	Навчально-науковий Північний локаль...	Доцент	26/10/2020	30/06/...
968	968	Сафранова Ірина Андріївна	Навчально-науковий Північний локаль...	Старший викладач	26/10/2020	30/06/...

Вид: Реєстр особових карток Рядки: 71

Рис 2.6 Підрозділи НУВГП

Інтерфейс модуля «Особові картки працівників» поділений на кілька блоків:

Меню: У верхній частині є опції меню, такі як "Реєстр", "Правка", "Вид" і "Сервіс".

Панель інструментів: Під меню розташовані різні іконки для дій, таких як додавання, редагування, видалення та пошук записів.

Фільтри та опції: Є опції для фільтрації записів за датою (15/12/2024), показу лише звільнених співробітників та функція пошуку.

Список підрозділів: Ліворуч знаходиться ієрархічний список підрозділів університету, таких як "Адміністрація", "Навчально-науковий інститут агроєкології" та інші.

Записи співробітників: Основний розділ праворуч відображає таблицю записів співробітників з колонками для номера картки, номера співробітника, прізвища, імені, по батькові, підрозділу, посади, дати прийому на роботу та дати звільнення.

НУВГП має ліцензію на 33 активних робочих місця в системі зареєстровано 55 підрозділів з загальною чисельністю 1938 осіб. Це дозволяє ефективно організувати роботу та забезпечити належний рівень обліку та управління персоналом.

### **3 РОЗДІЛ. Удосконалення інформаційної системи бухгалтерського та управлінського обліку НУВГП на основі ISpro**

#### **3.1 Стратегія впровадження системи ISpro в НУВГП**

Прийняття стратегічного рішення перейти на ISpro було зумовлено кількома вагомими причинами. По-перше, гнучкість налаштувань ISpro дозволяє адаптувати систему до унікальних бізнес-процесів, що було складним завданням у попередній системі 1С. Зокрема, ми змогли легко налаштувати модулі, що дозволило оптимізувати роботу.

По-друге, наявність потужного API в ISpro відкриває перед нами широкі можливості для інтеграції з іншими системами, які ми використовуємо. Ми плануємо інтегрувати ISpro з ЄДБО, що дозволить нам створити єдиний інформаційний простір та автоматизувати ще більше процесів.

Крім того, високий рівень локалізації ISpro, включаючи підтримку українського законодавства та мови, значно спростив процес міграції та подальшого використання системи.

Успішне впровадження інформаційної системи, такої як ISpro, вимагає ретельної підготовки та чітко визначеної стратегії. Цей розділ описує ключові етапи та підходи до впровадження ISpro в НУВГП, враховуючи специфіку університету.

##### **1. Аналіз поточного стану та визначення цілей**

Першим кроком є проведення детального аналізу існуючої системи обліку в НУВГП. Цей аналіз повинен охоплювати:

Оцінка існуючої IT-інфраструктури: Аналіз апаратного та програмного забезпечення, мережевої інфраструктури, каналів зв'язку.

Аналіз бізнес-процесів: Детальне вивчення процесів бухгалтерського та управлінського обліку, документообігу, формування звітності.

Визначення потреб користувачів: Збір інформації від працівників бухгалтерії, фінансового відділу, керівництва університету щодо їхніх потреб та очікувань від нової системи.

SWOT-аналіз: Визначення сильних та слабких сторін існуючої системи, а також можливостей та загроз, пов'язаних з впровадженням ISpro.

На основі проведеного аналізу необхідно сформулювати чіткі цілі впровадження ISpro, наприклад:

Автоматизація рутинних облікових операцій.

Підвищення оперативності та достовірності облікової інформації.

Забезпечення інтеграції між різними підрозділами університету.

Покращення контролю за використанням бюджетних коштів.

Підвищення ефективності управління фінансовими ресурсами.

## 2. Планування проекту впровадження

На цьому етапі розробляється детальний план проекту впровадження, який включає:

Формування команди проекту: Визначення відповідальних осіб з боку університету та компанії-постачальника ISpro.

Розробка графіку виконання робіт: Визначення термінів виконання кожного етапу проекту.

Бюджетування проекту: Оцінка витрат на придбання ліцензій, обладнання, навчання персоналу, послуги з впровадження та підтримки.

План навчання користувачів: Розробка програми навчання для працівників університету, які будуть використовувати ISpro.

План міграції даних: Визначення стратегії перенесення даних з існуючих систем в ISpro.

План тестування та введення в експлуатацію: Визначення процедур тестування системи та поетапного введення її в експлуатацію.

## 3. Етапи впровадження

Рекомендується поетапне впровадження ISpro, що дозволяє мінімізувати ризики та забезпечити більш плавний перехід на нову систему. Типові етапи впровадження включають:

Підготовка: Встановлення та налаштування програмного забезпечення, підготовка IT-інфраструктури, навчання ключових користувачів.

Міграція даних: Перенесення даних з існуючих систем в ISpro.

Тестування: Проведення комплексного тестування системи для перевірки її функціональності та відповідності вимогам.

Пілотне впровадження: Введення системи в експлуатацію в окремому підрозділі університету для апробації та виявлення можливих проблем.

Повне впровадження: Поступове розширення використання системи на всі підрозділи університету.

Підтримка та супровід: Забезпечення технічної підтримки та супроводу системи після введення в експлуатацію.

#### 4. Управління змінами

Впровадження нової інформаційної системи завжди пов'язане зі змінами в робочих процесах та звичках працівників. Важливо приділити увагу управлінню змінами, що включає:

Комунікація: Інформування працівників про цілі та переваги впровадження ISpro.

Залучення користувачів: Активне залучення працівників до процесу впровадження, врахування їхніх побажань та пропозицій.

Навчання та підтримка: Забезпечення якісного навчання та постійної підтримки користувачів.

#### 5. Оцінка ефективності

Після впровадження ISpro необхідно провести оцінку ефективності системи, використовуючи визначені на початку проекту критерії. Оцінка може включати:

Аналіз показників продуктивності роботи.

Оцінку рівня задоволеності користувачів.

Аналіз економічної ефективності впровадження.

### **3.2 Створення модуля електронної розсилки відомостей про заробітну плату ISpro**

Створення модуля електронної розсилки відомостей про заробітну плату в системі ISpro включає наступні ключові етапи:

1. Аналіз вимог: Визначення потреб користувачів та вимог до функціональності модуля. Це включає визначення формату відомостей, частоти розсилки та способів доставки (електронна пошта, SMS тощо).

2. Проектування модуля: Розробка архітектури модуля, включаючи базу даних, інтерфейс користувача та інтеграцію з існуючими системами обліку заробітної плати.

3. Розробка: Написання коду для модуля, включаючи функції для генерації відомостей про заробітну плату, формування повідомлень та їх відправку.

4. Тестування: Перевірка модуля на відповідність вимогам, виявлення та виправлення помилок. Це включає тестування функціональності, безпеки та продуктивності.

5. Впровадження: Інтеграція модуля в існуючу систему ISpro, навчання користувачів та налаштування параметрів розсилки.

6. Підтримка та оновлення: Забезпечення технічної підтримки користувачів, виправлення помилок та оновлення модуля для покращення його функціональності.

Аналіз вимог включає визначення потреб користувачів та вимог до функціональності модуля:

1. Формат відомостей:

- Визначення, які дані повинні бути включені у відомості про заробітну плату (наприклад, сума заробітної плати, податки, відрахування, бонуси тощо).

- Вибір формату файлу для розсилки (наприклад, PDF, Excel, текстовий файл).

## 2. Частота розсилки:

- Визначення, як часто повинні відправлятися відомості (щомісяця, щотижня, після кожної виплати заробітної плати).

- Налаштування графіку розсилки, щоб забезпечити своєчасне отримання інформації користувачами.

## 3. Способи доставки:

- Вибір каналів доставки відомостей (електронна пошта, SMS, внутрішня система повідомлень).

- Забезпечення безпеки передачі даних, щоб конфіденційна інформація не потрапила до сторонніх осіб.

## 4. Інтерфейс користувача:

- Розробка зручного та інтуїтивно зрозумілого інтерфейсу для налаштування параметрів розсилки.

- Можливість користувачів самостійно налаштовувати параметри розсилки відповідно до своїх потреб.

## 5. Інтеграція з існуючими системами:

- Забезпечення сумісності модуля з існуючими системами обліку заробітної плати.

- Налаштування автоматичного оновлення даних для забезпечення актуальності інформації.

Таблиця 3.1

### Вимоги до модуля розсилки

Категорія	Опис
Формат відомостей	Включення даних про суму заробітної плати, податки, відрахування, бонуси, Формат файлу: PDF, Excel.
Частота розсилки	Щомісяця, щотижня, після кожної виплати

	заробітної плати. Налаштування графіку розсилки
Способи доставки	Електронна пошта, SMS, внутрішня система повідомлень. Забезпечення передачі даних.
Інтерфейс користувача	Зручний та інтуїтивно зрозумілий інтерфейс для налаштування параметрів розсилки. Можливість самостійного налаштування параметрів.
Інтеграція з існуючими системами	Сумісність з існуючими системами обліку заробітної плати. Автоматичне оновлення даних.

2. Проектування модуля та Розробку коду будемо здійснювати у конструктор ISpro . Модуль дизайнер звітів дозволяє автоматизувати написання коду написання коду для модуля, включаючи функції для генерації відомостей про заробітну плату, формування повідомлень та їх відправку

Модуль дизайнера представлений на рисунку 3.1.

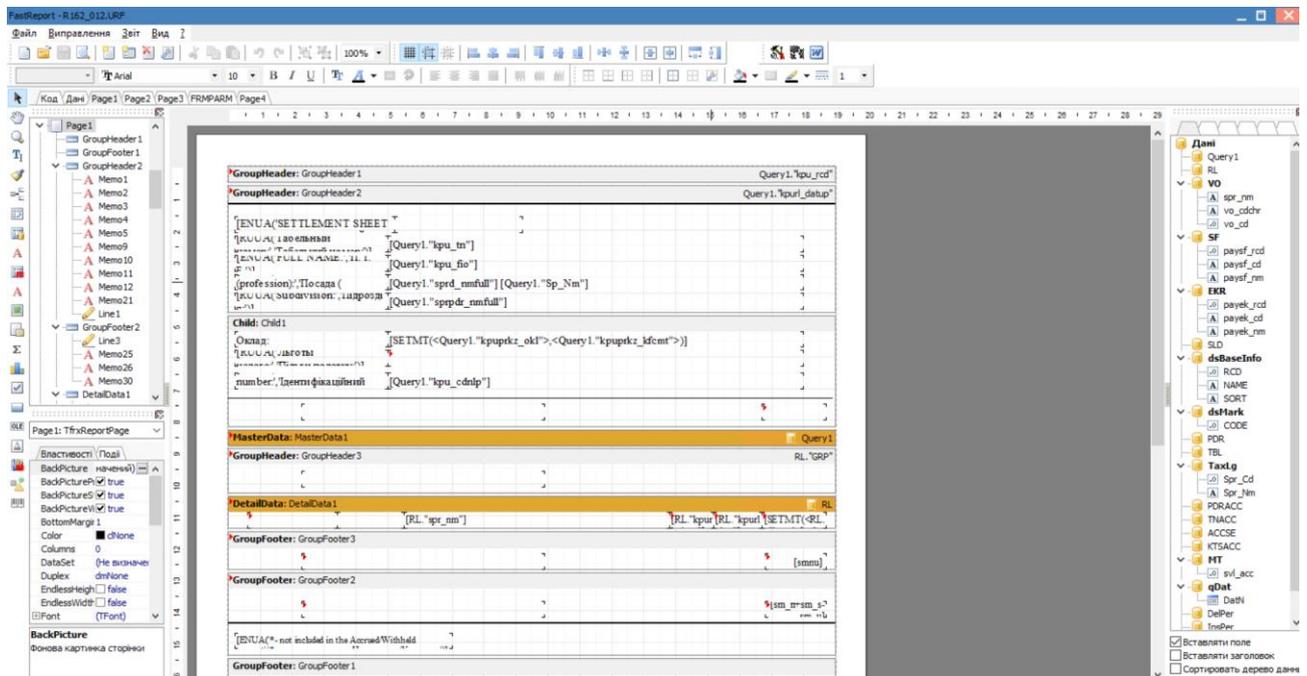


Рис 3.1. Перша сторінка звіту

## Фрагмент коду модуля розсилки представлений на рисунку 3.1 Додаток Б

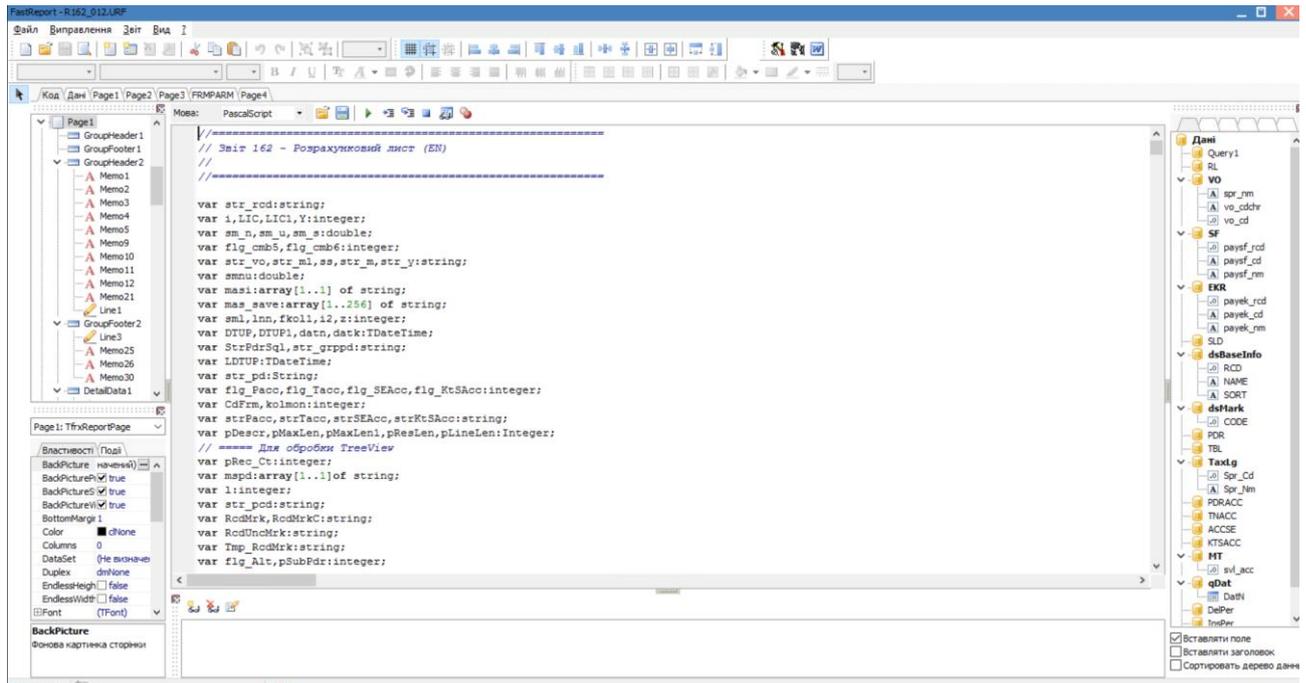


Рис 3.2 Фрагмент коду модуля розсилки

В результаті кожен працівник університету отримує типовий розрахунковий лист представлений на Рис 3.4 типовий лист

Розсилка розрахункових листів електронною поштою має наступні значні переваги:

**Економія часу та ресурсів:** Автоматизована розсилка дозволяє зменшити витрати на папір, друк та поштові послуги. Це також зменшує час, необхідний для підготовки та відправки розрахункових листів. До запровадження автоматизованої розсилки Бухгалтерія університету що місячно друкувала біль ніж дві тисячі розрахункових листів, тобто в середньому ~2 пачки паперу якісного друку. Тобто в рік витрачалось близько п'ятдесяти тисяч грн на друк

**Зручність для працівників:** Працівники можуть отримувати свої розрахункові листи безпосередньо на електронну пошту, що дозволяє їм швидко та зручно переглядати інформацію про заробітну плату.

**Безпека даних:** Електронна пошта дозволяє забезпечити конфіденційність та безпеку передачі даних за допомогою шифрування та захищених каналів зв'язку.

**Швидкість доставки:** Розрахункові листи доставляються миттєво, що дозволяє працівникам отримувати інформацію про заробітну плату вчасно та без затримок.

**Екологічність:** Використання електронної пошти зменшує споживання паперу та інших ресурсів, що позитивно впливає на навколишнє середовище.

**Легкість зберігання та доступу:** Електронні розрахункові листи легко зберігати та архівувати, що дозволяє працівникам швидко знайти необхідну інформацію у разі потреби.

```

<meta http-equiv="Content-Type" content="text/html; charset=windows-1251">
</head>
<body bgcolor="#FFFFFF" text="#000000">
<title>
РОЗРАХУНКОВИЙ ЛИСТ ЗА ЛИСТОПАД 2024
</title>
<h3>РОЗРАХУНКОВИЙ ЛИСТ ЗА ЛИСТОПАД 2024</h3>
<pre>
<style='line-height:15pt'>
<font size=3>Табельний номер...: 1229</font>
<font size=3>П.І.В.....: Василів Володимир Богданович</font>
<font size=3>Посада.....: Доцент</font>
<font size=3>Підрозділ.....: Кафедра комп'ютерних технологій та економічної кібернетики</font>
<font size=3>Оклад.....: 12'128.97</font>
<font size=3>Пільги податку.....: Відсутні</font>
<font size=3>Ідентифікаційний №: 2711602876</font>
</pre>
<hr></hr>
<font size=4 face="Arial">БОРГ ПІДПРИЄМСТВА НА ЛИСТОПАД 2024:0.00</font><hr></hr>
<table cellspacing=10>
<tr><td align=right>11/24</td><td align=right> 1</td><td align=left>Основний оклад</td><td align=right>26</td><td align=right>156.00</td><td align=right>6'064.48</td></tr>
<tr><td align=right></td><td align=right> 101</td><td align=left>Надбавка за вислугу років</td><td align=right>26</td><td align=right>156.00</td><td align=right>1'819.34</td></tr>
<tr><td align=right></td><td align=right> 105</td><td align=left>Доплата за вчене звання</td><td align=right>26</td><td align=right>156.00</td><td align=right>1'516.12</td></tr>
<tr><td align=right></td><td align=right> 106</td><td align=left>Доплата за науковий ступінь</td><td align=right>26</td><td align=right>156.00</td><td align=right>909.67</td></tr>
<TR><TH ALIGN=LEFT COLSPAN=5>ВСЬОГО НАРАХОВАНО:</TH><TH ALIGN=RIGHT>10'309.61</TH></TR>
<tr><td align=right>11/24</td><td align=right>400</td><td align=left>Податок з доходів</td><td align=right></td><td align=right></td><td align=right>1'855.73</td></tr>
<tr><td align=right></td><td align=right>401</td><td align=left>Військовий збір</td><td align=right></td><td align=right></td><td align=right>154.64</td></tr>
<tr><td align=right></td><td align=right> 801</td><td align=left>Заробітна плата за 1-у половину місяця</td><td align=right></td><td align=right></td><td align=right>4'149.62</td></tr>
<tr><td align=right></td><td align=right> 820</td><td align=left>Заробітна плата за 2-у половину місяця</td><td align=right></td><td align=right></td><td align=right>4'149.62</td></tr>
<TR><TH ALIGN=LEFT COLSPAN=5>УТРИМАНО:</TH><TH ALIGN=RIGHT>10'309.61</TH></TR>
</table>
<font size=4 face="Arial">БОРГ ПІДПРИЄМСТВА НА ГРУДЕНЬ 2024:</font><font size=4 face="Arial">0.00</font>
<hr></hr>
</body>

```

Рис. 3.3. Структура розрахункового листа

## РОЗРАХУНКОВИЙ ЛИСТ ЗА ЛИСТОПАД 2024

Табельний номер....: 1229  
 П.І.Б.....: Василів Володимир Богданович  
 Посада.....: Доцент  
 Підрозділ.....: Кафедра комп'ютерних технологій та економічної кібернетики  
 Оклад.....: 12'128.97  
 Пільги податку....: Відсутні  
 Ідентифікаційний №: 2711602876

### БОРГ ПІДПРИЄМСТВА НА ЛИСТОПАД 2024:0.00

11/24	1	Основний оклад	26	156.00	6'064.48
	101	Надбавка за вислугу років	26	156.00	1'819.34
	105	Доплата за вчене звання	26	156.00	1'516.12
	106	Доплата за науковий ступінь	26	156.00	909.67

Рис. 3.4 Приклад розрахункового листа

### 3.3 Розробка функціонального звіту реєстр соціальних пільг студентів на базі ISpro

**FastReport** - це потужний інструмент для створення звітів, який інтегрується з системою ISpro. Він дозволяє користувачам створювати, редагувати та генерувати різноманітні звіти на основі даних з бази даних ISpro. Ось кілька ключових можливостей FastReport для ISpro:

**Дизайнер звітів:** Візуальний інтерфейс для створення та налаштування звітів. Користувачі можуть додавати різні елементи, такі як таблиці, графіки, текстові поля та інші компоненти, щоб створити звіт відповідно до своїх потреб.

**Підтримка різних джерел даних:** FastReport може працювати з різними джерелами даних, включаючи бази даних, масиви, файли та інші джерела. Це дозволяє користувачам легко інтегрувати дані з різних систем.

**Гнучкість налаштувань:** Користувачі можуть налаштовувати параметри звітів, такі як діапазони дат, фільтри та інші параметри, щоб отримати необхідну інформацію. Це робить звіти більш точними та релевантними.

**Автоматизація:** FastReport дозволяє автоматизувати процес створення та розсилки звітів. Це зменшує ручну роботу та підвищує ефективність роботи.

**Підтримка скриптів:** FastReport підтримує використання скриптів для додаткової обробки даних та налаштування звітів. Це дозволяє користувачам реалізувати складні логічні операції та автоматизацію.

Порядок розробки звіту у FastReport

#### **Підготовка даних:**

Визначення джерел даних, які будуть використовуватися у звіті (бази даних, файли, масиви тощо).

Реєстрація джерел даних у FastReport.

#### **Створення шаблону звіту:**

Відкриття FastReport Designer.

Створення нового звіту або відкриття існуючого шаблону.

Додавання необхідних компонентів (таблиці, графіки, текстові поля тощо) на сторінку звіту.

#### **Налаштування параметрів звіту:**

- Визначення параметрів звіту, таких як діапазони дат, фільтри та інші налаштування.
- Додавання діалогових вікон для введення параметрів користувачем, якщо це необхідно.
- Розробка логіки звіту:
- Використання скриптів для реалізації складної логіки обробки даних.
- Налаштування обробки подій для компонентів звіту.
- Попередній перегляд та тестування:
- Виконання попереднього перегляду звіту, щоб переконатися, що всі дані відображаються правильно.

- Перевірка роботи параметрів та логіки звіту.
- Експорт та розповсюдження:
- Налаштування параметрів експорту звіту у потрібний формат (PDF, Excel, HTML тощо).
- Збереження звіту та розповсюдження його серед користувачів.

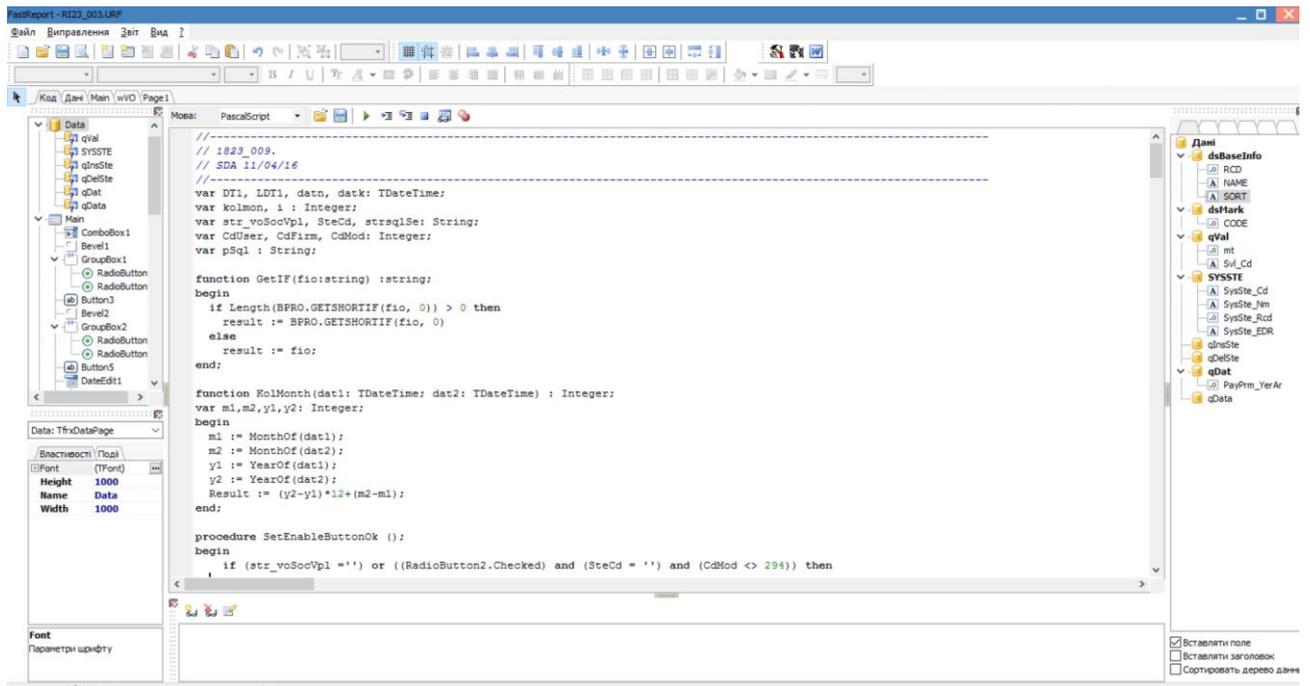


Рис 3.5 Вікно коду у Fastreport

На рис 3.5 показано інтерфейс FastReport, який використовується для створення та управління звітами. Інтерфейс українською мовою. Головне вікно відображає редактор коду PascalScript з написаним кодом зокрема: коментарями змінними, функціями та процедурами. Ліва панель показує дерево структури даних та компонентів, що використовуються у звіті, а права панель показує інше дерево структури полів даних.

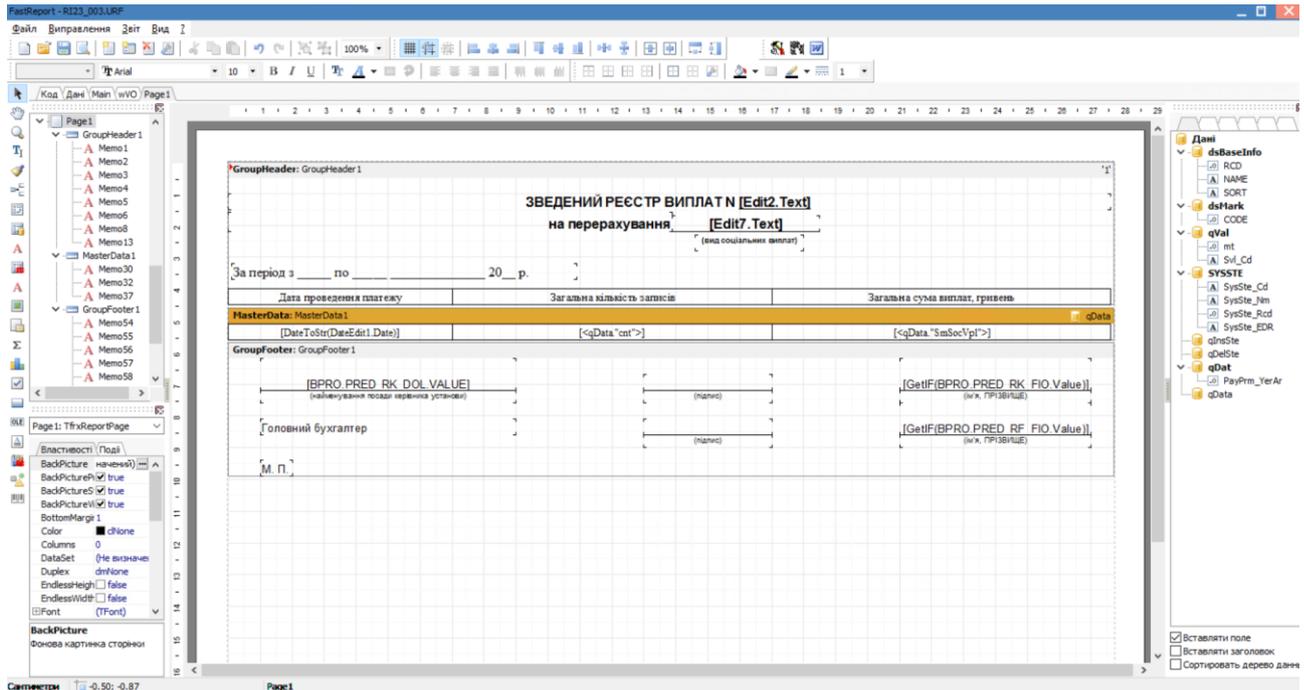


Рис 3.6 Зведений реєстр виплат

Структура зведеного реєстру пільгових виплат має чітку та логічну структуру, що дозволяє легко орієнтуватися в інформації. Це особливо важливо для зведених реєстрів виплат, де необхідно швидко знайти потрібні дані. Використання заповнювачів (placeholders) дозволяє автоматично вставляти динамічні дані, що забезпечує гнучкість та можливість повторного використання звіту для різних періодів та типів виплат. Автоматизація процесу створення звітів зменшує кількість ручної роботи та знижує ризик помилок, що підвищує загальну ефективність роботи.

### 3.4 Розробка модуля API для ISpro

Розробка модуля API для ISpro є важливим кроком для розширення функціональності системи та інтеграції її з іншими програмними продуктами. API (Application Programming Interface) дозволяє програмам обмінюватися даними та функціями, що відкриває можливості для автоматизації процесів, створення звітів, розробки мобільних додатків та інших корисних інструментів.

### *Визначення потреб та цілей*

Перш ніж розпочинати розробку API, необхідно чітко визначити потреби та цілі його створення:

Які функції ISpro потрібно зробити доступними через API? (Наприклад, отримання даних про працівників, нарахування заробітної плати, формування звітів, управління документами).

Які системи будуть інтегруватися з ISpro через API? (Наприклад, системи управління кадрами, CRM, системи електронного документообігу).

Які вимоги до безпеки та продуктивності API? (Наприклад, авторизація, шифрування, обмеження кількості запитів).

Які формати даних будуть використовуватись? (Наприклад, JSON, XML).

### *Аналіз можливостей ISpro*

Важливо проаналізувати наявні можливості ISpro для розробки API. Згідно з документацією ISpro, система має REST API, що використовує протокол HTTP та формат даних JSON. Це сучасний та зручний підхід, який дозволяє легко інтегрувати ISpro з різними платформами.

### *Проектування API*

На етапі проектування необхідно визначити:

Кінцеві точки API (endpoints): URL-адреси, за якими будуть доступні окремі функції API.

Методи HTTP (GET, POST, PUT, DELETE): Для виконання різних операцій з даними (отримання, створення, оновлення, видалення).

Формат запитів та відповідей: Структура даних, які передаються між клієнтом та сервером.

Коди стану HTTP: Для інформування про результат виконання запиту (наприклад, 200 OK, 400 Bad Request, 500 Internal Server Error).

Механізми аутентифікації та авторизації: Для захисту API від несанкціонованого доступу.

### *Розробка API*

Розробка API включає написання коду, який обробляє запити та повертає відповіді. Для цього можна використовувати різні мови програмування та фреймворки, залежно від потреб та можливостей ISpro. Важливо дотримуватися принципів RESTful API та забезпечити якісне документування коду.

### *Тестування API*

Після розробки необхідно провести тестування API для перевірки його коректної роботи. Тестування повинно включати:

Перевірку всіх кінцевих точок та методів HTTP.

Перевірку форматів запитів та відповідей.

Перевірку механізмів аутентифікації та авторизації.

Перевірку обробки помилок.

Тестування продуктивності та безпеки.

Для тестування API можна використовувати спеціальні інструменти, такі як Postman або Swagger UI.

### *Документування API*

Важливим етапом є створення документації для API, яка описує його функціональність, кінцеві точки, формати даних, механізми аутентифікації та приклади використання. Для документування REST API часто використовується специфікація OpenAPI (Swagger).

### *Впровадження та підтримка*

Після тестування та документування API можна впроваджувати його в експлуатацію. Важливо забезпечити технічну підтримку та своєчасне оновлення API.

Використання API ISpro для розробки модуля електронної розсилки відомостей про заробітну плату

Розроблений API можна використовувати для створення модуля електронної розсилки відомостей про заробітну плату, описаного в

попередньому розділі. Модуль може використовувати API для отримання даних про заробітну плату працівників та автоматичної генерації розрахункових листів у форматі PDF.

Ось приклад коду, який реалізує функцію отримання відомостей про заробітну плату:

Python

```
import requests
```

```
def get_salary_info(api_key, base_url, employee_id=None, period=None):
```

```
    """
```

Отримує відомості про заробітну плату з ISpro API.

Args:

api\_key: Ключ API для автентифікації.

base\_url: Базова URL-адреса API ISpro.

employee\_id: Ідентифікатор працівника (опціонально).

period: Період, за який потрібно отримати дані (опціонально, наприклад, "2023-10").

Returns:

Дані про заробітну плату у форматі JSON або None у випадку помилки.

```
    """
```

```
    url = f'{base_url}/salary' # Припускаємо, що URL для зарплати
/salary
```

```
    headers = {
```

```
        "Authorization": f"Bearer {api_key}",
```

```
        "Content-Type": "application/json"
```

```
    }
```

```
    params = {}
```

```

if employee_id:
    params["employee_id"] = employee_id
if period:
    params["period"] = period

try:
    response = requests.get(url, headers=headers, params=params)
    response.raise_for_status()
    return response.json()
except requests.exceptions.RequestException as e:
    print(f"Помилка запиту: {e}")
    return None

# Приклад використання
api_key = "ВАШ_КЛЮЧ_API" # Замініть на ваш реальний ключ
API
base_url = "https://api.ispro.com/v1" # Замініть на реальну базову
URL API

# Отримати дані про зарплату для всіх працівників за весь час
all_salary_data = get_salary_info(api_key, base_url)
if all_salary_data:
    print("Дані про зарплату (всі працівники, весь час):",
all_salary_data)

# Отримати дані про зарплату для працівника з ID 123 за жовтень
2023

salary_data_employee_123 = get_salary_info(api_key, base_url,
employee_id="123", period="2023-10")
if salary_data_employee_123:

```

```
print("Дані про зарплату (працівник 123, жовтень 2023):",
salary_data_employee_123)
```

Пояснення коду:

```
get_salary_info(api_key, base_url, employee_id=None, period=None):
```

Ця функція надсилає GET-запит до API ISpro для отримання даних про заробітну плату.

`url = f"{base_url}/salary"`: Ми припускаємо, що кінцева точка API для отримання даних про заробітну плату - /salary.

`headers`: Містить заголовки запиту, включаючи ключ API для автентифікації та тип контенту (JSON).

`params`: Словник з параметрами запиту. Ми додаємо `employee_id` та `period`, якщо вони вказані.

`requests.get(...)`: Виконує GET-запит до API.

`response.raise_for_status()`: Перевіряє, чи код відповіді HTTP знаходиться в діапазоні 200-299 (успішний запит). Якщо ні, то генерується виключення.

`response.json()`: Повертає дані у форматі JSON.

### 3.5 Скрипти користувача ISpro WEB

Скрипти користувача ISpro WEB призначені для розширення функціональності веб-інтерфейсу ISpro за допомогою користувацьких скриптів. Це дозволяє адаптувати систему під специфічні потреби організації та автоматизувати рутинні завдання.

ISpro WEB надає можливість використання користувацьких скриптів для розширення та кастомізації функціональності веб-інтерфейсу. Скрипти дозволяють автоматизувати дії користувача, змінювати поведінку елементів інтерфейсу, додавати нові функції та інтегрувати систему з іншими веб-сервісами.

Користувацькі скрипти в ISpro WEB можуть виконувати наступні завдання:

Автоматизація рутинних операцій: Автоматичне заповнення полів, виконання послідовності дій, перевірка даних.

Зміна поведінки елементів інтерфейсу: Приховування або відображення елементів, зміна їх стилю, додавання обробників подій.

Валідація даних на клієнті: Перевірка коректності введених даних перед відправкою на сервер.

Інтеграція з зовнішніми веб-сервісами: Отримання даних з інших веб-сайтів або сервісів, відправка даних на зовнішні сервери.

Створення користувацьких звітів та візуалізацій: Обробка даних, отриманих з ISpro, та відображення їх у зручному форматі.

Технології, що використовуються

Для написання скриптів користувача в ISpro WEB використовується мова програмування JavaScript. JavaScript є стандартною мовою для веб-розробки та підтримується всіма сучасними браузерами. Для роботи з DOM (Document Object Model) та API ISpro можна використовувати стандартні JavaScript-методи та об'єкти.

Розробка скриптів користувача включає наступні етапи:

Визначення завдання: Чітко сформулювати завдання, яке повинен виконувати скрипт.

Аналіз інтерфейсу ISpro: Вивчити структуру HTML-сторінки, з якою буде працювати скрипт, визначити необхідні елементи та їх властивості.

Написання коду скрипту: Написати код на JavaScript, який виконує поставлене завдання.

Тестування скрипту: Перевірити роботу скрипту в різних браузерах та в різних умовах.

Впровадження скрипту: Завантажити скрипт в ISpro WEB або додати його на відповідну сторінку.

## Приклад скрипту Зміна кольору фону елемента

JavaScript

```
// Знайти елемент за ID
```

```
var element = document.getElementById("myElement");
```

```
// Перевірити, чи елемент існує
```

```
if (element) {
```

```
    // Змінити колір фону
```

```
    element.style.backgroundColor = "lightgreen";
```

```
} else {
```

```
    console.log("Елемент з ID 'myElement' не знайдено.");
```

```
}
```

// Приклад використання API ISpro (потрібно дивитись документацію ISpro)

```
// ISpro.API.get('/api/employees', function(data) {
```

```
//   console.log(data);
```

```
// });
```

Приклад скрипту Блокування полів

Модуль: Документообіг та управління процесами / Документи

Шаблон: Користувальницький Вхідний лист системний шаблон.

Поле На №, якщо значення даного поля буде більше 100, то поле номер буде заблоковано і якщо поле буде менше 100, то поле буде доступне для редагування.

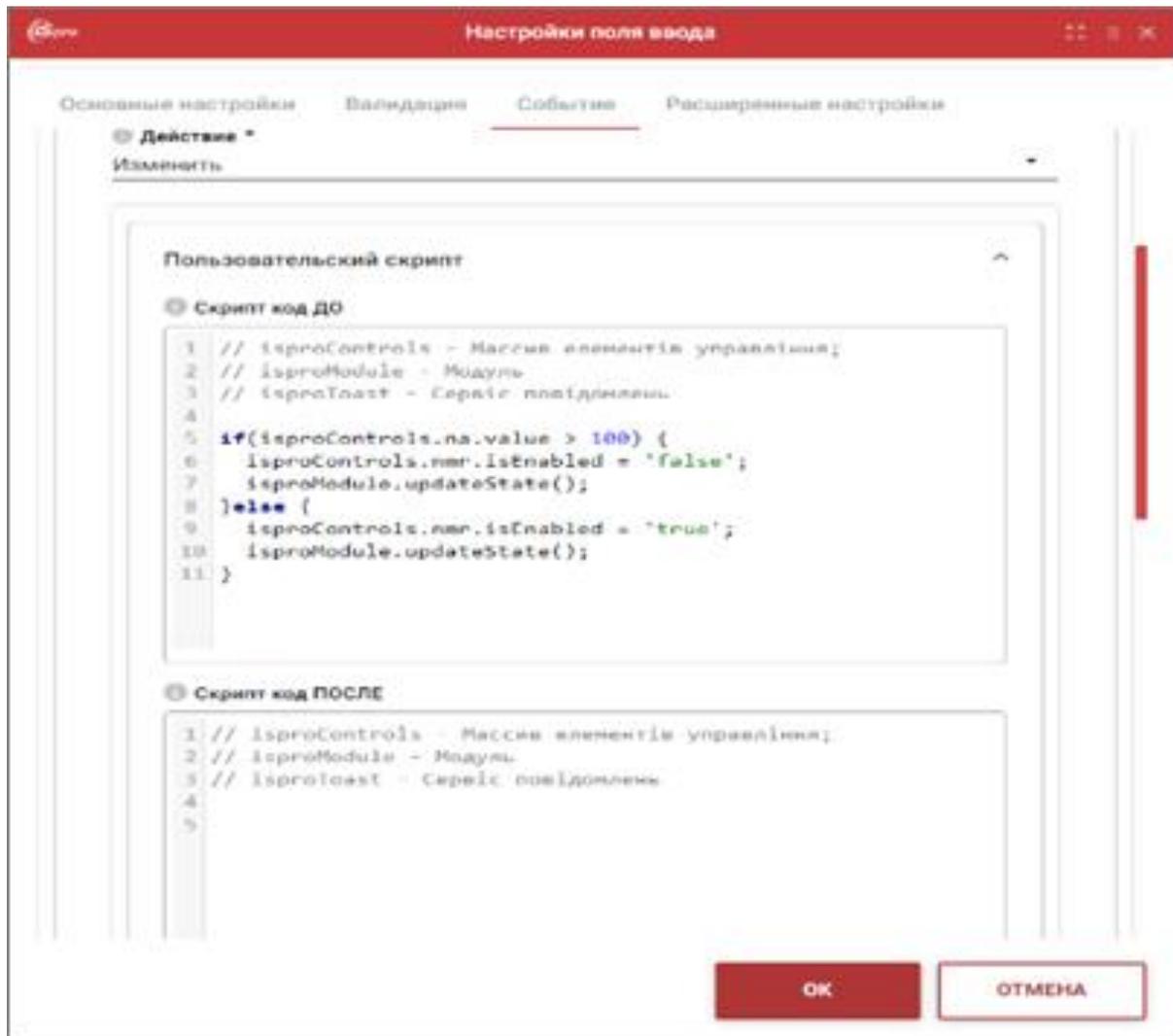


Рис. 3.7 Налаштування скрипта

Текст скрипта:

```

if(isproControls.na.value > 100) {
isproControls.nmr.isEnabled = 'false';
isproModule.updateState();
}else {
isproControls.nmr.isEnabled = 'true';
isproModule.updateState();
}

```

### Приклад скрипта Складові значення полів

Модуль: Документообіг та управління процесами/ Документи

Шаблон: Користувача Вхідний лист(номер шаблону 4 код 1)

Суть роботи скрипта в тому, що при зміні значення комбо **Від кого** необхідно сформуванати префікс для поля номер документа.

Префікс формуватимемо за першими літерами ПІБ.

Приклад цього скрипту:

```

1 Скрипт код ДО
5 let recipient = isproModule.model.ot.name;
6
7 if(isproControls.nmr.value == null || isproControls.nmr.value == '') {
8   isproControls.nmr = recipient.split(" ").map(s => s[0]).join("") + '-' + '';
9   isproToast.showMessage('Пользовательский скрипт №1', 'Скрипт отработал, первый блок', 1);
10 }else if (isproControls.nmr.value.includes('-')){
11   isproControls.nmr = isproControls.nmr.value.split('-')[1];
12   isproControls.nmr = recipient.split(" ").map(s => s[0]).join("") + '-' + isproControls.nmr.value;
13   console.log(isproControls);
14   isproControls.dat.isEnabled = 'false';
15   isproToast.showMessage('Пользовательский скрипт №2', 'Скрипт отработал, второй блок', 2);
16 }else{
17   isproControls.nmr = recipient.split(" ").map(s => s[0]).join("") + '-' + isproControls.nmr.value;
18   isproToast.showMessage('Пользовательский скрипт №3', 'Скрипт отработал, третий блок', 3);

```

Текст скрипту:

```

let recipient = isproModule.model.ot.name;
if(isproControls.nmr.value == null || isproControls.nmr.value == "") {
  isproControls.nmr = recipient.split(" ").map(s => s[0]).join("") + '-' + "";
  isproToast.showMessage('Скрипт користувача 1', 'Скрипт відпрацював,
перший блок', 1);
} else if (isproControls.nmr.value.includes('-')){
  isproControls.nmr = isproControls.nmr.value.split('-')[1];
  isproControls.nmr = recipient.split(" ").map(s => s[0]).join("") + '-' +
isproControls.nmr.value;
  console.log(isproControls);
  isproControls.dat.isEnabled = 'false';
  isproToast.showMessage('Скрипт користувача 2', 'Скрипт відпрацював,
другий блок', 2);
} else{

```

```

isproControls.nmr = recipient.split(" ").map(s => s[0]).join("") + '-' +
isproControls.nmr.value;
isproToast.showMessage('Скрипт користувача 3', 'Скрипт відпрацював,
третій блок', 0);
}

```

### Впровадження та використання скриптів

ISpro WEB, як правило, надає механізм для завантаження та керування користувацькими скриптами. Зазвичай це робиться через адміністративний інтерфейс або шляхом додавання скриптів на окремі сторінки. Детальна інформація про впровадження скриптів повинна бути в документації ISpro.

### Безпека скриптів

Важливо пам'ятати про безпеку при використанні користувацьких скриптів. Не слід виконувати скрипти з ненадійних джерел. Також необхідно обмежувати права доступу користувачів до завантаження та виконання скриптів.

### Переваги використання скриптів користувача

Гнучкість та кастомізація: Адаптація ISpro під специфічні потреби організації.

Автоматизація рутинних завдань: Збільшення продуктивності та зменшення кількості помилок.

Розширення функціональності: Додавання нових можливостей, які не передбачені стандартною функціональністю ISpro.

### Обмеження

Необхідні знання JavaScript та HTML.

Скрипти можуть залежати від структури HTML-сторінок ISpro, яка може змінюватися з оновленнями системи. Тому скрипти потрібно буде перевіряти та оновлювати після оновлень ISpro.

## ВИСНОВКИ

У дипломній роботі проведено аналіз стану інформаційної системи бухгалтерського та управлінського обліку Національного університету водного господарства та природокористування (НУВГП) та розроблено пропозиції щодо її вдосконалення на основі програмного забезпечення ISpro.

Дослідження показало, що існуюча система обліку в НУВГП має ряд недоліків, зокрема:

Відсутність єдиної інформаційної платформи для управлінського обліку.

Значна частина інформації обробляється вручну, що призводить до помилок та збільшує трудомісткість облікових процесів.

Більшість документів зберігаються у паперовому вигляді, що уповільнює доступ до даних та збільшує витрати на забезпечення друку.

Дублювання ключового поля при одночасній роботі кількох користувачів у одній таблиці та тривалий час обробки документів.

Враховуючи виявлені проблеми, в дипломній роботі обґрунтовано вибір ISpro як платформи для автоматизації обліку в НУВГП. ISpro – це сучасна інформаційна система, яка відповідає вимогам бюджетного законодавства та дозволяє автоматизувати різні ділянки обліку, включаючи бухгалтерський та управлінський.

В рамках дипломної роботи розроблено два модулі для ISpro:

Модуль електронної розсилки відомостей про заробітну плату, який дозволяє автоматизувати процес інформування працівників про нараховану заробітну плату.

Функціональний звіт "Реєстр соціальних пільг студентів", який надає інформацію про студентів, які мають право на соціальні пільги.

Впровадження запропонованих рішень дозволить:

Підвищити ефективність облікових процесів в НУВГП.

Зменшити трудомісткість обліку та кількість помилок.

Прискорити доступ до інформації та покращити якість управлінських рішень.

Зменшити витрати на забезпечення обліку.

Отримані навички у проектуванні звітів FastReport, створенні автоматизованих розсилок дозволили забезпечити ефективне рішення для організації. Такий підхід допомагає економити ресурси, покращує обслуговування працівників і робить систему більш ефективною.

## ВИКОРИСТАНІ ДЖЕРЕЛА

1. Бенько М.М. Інформаційні системи і технології в бухгалтерському обліку: монографія. – Київ: Київ. нац. торг.-екон. ун-т, 2010. – 336 с. ISBN 978-966-629-422-0.).
2. Бегоцкая Г.К. Обработка учетной информации с использованием вычислительной техники / Г.К. Бегоцкая, Э.А. Умнова – М. : Финансы, 1977. – 70 с.
3. Сусіденко В. Т. Інформаційні системи і технології в обліку: навч. посіб. / В. Т. Сусіденко. – Київ: Центр учбової літератури, 2016. – 224 с. ISBN 978-617-673-440-6.
4. Завгородний В.П. Автоматизация бухгалтерского учета, контроля, анализа и аудита : монография / В.П. Завгородний. – К. : АСК, 1998. – 768 с.
5. Принципи роботи в системі: веб-сайт. URL: <https://doc.ispro.ua/ua/pdf/init.pdf> (дата звернення: 11.12.2024).
6. Писаревська Т.А. Інформаційні системи обліку та аудиту : навч. посіб. / Т.А. Писаревська. – К. : КНЕУ, 2004. – 369 с.
7. Білуха М.Т. Основні напрями розвитку обліку та контролю за електронною технологією в XXI столітті / М.Т. Білуха // Вісн. КНТЕУ. – 2005. – № 3. – С. 13–17.
8. Річний звіт ректора 2023: веб-сайт. URL: <https://nuwm.edu.ua/publicna-informacija/normativni-dokumenti/richnij-zvit-rektora>
9. Проект Закону про заборону продажу на території України програмного забезпечення, для податкового та бухгалтерського обліку, при створенні якого використано програмний код, написаний мовою країни-агресора: веб-сайт. URL: <https://itd.rada.gov.ua/billInfo/Bills/Card/43054>

Додатки

Додаток А

## Експертний висновок



## ЕКСПЕРТНИЙ ВИСНОВОК

**Захищений програмний комплекс «ІС-ПРО» версії 8.XX.XXX**  
виробництва ТОВ «Інтелект-сервіс»

Зареєстровано в Адміністрації Державної служби спеціального зв'язку та захисту інформації України  
"8" жовтня 2020 р. за № 1173  
Дійсний до "8" жовтня 2023 р.

Результати експертизи свідчать, що захищений програмний комплекс «ІС-ПРО» версії 8.XX.XXX,  
(назва засобу ТЗІ)

який наданий на експертизу **ТОВ «Інтелект-сервіс»**,  
найменування організації (підприємства,  
**м. Київ, вул. Кирилівська, 102**,  
установи) – власника (розпорядника), місцезнаходження)

**відповідає**  
(відповідає/не відповідає)

вимогам нормативних документів з технічного захисту інформації в обсязі функцій, зазначених у документі «Захищений програмний комплекс «ІС-ПРО» версії 8.XX.XXX. Технічне завдання».

Вимоги до умов експлуатації та сфери використання об'єкта експертизи визначені у відповідному розділі Експертного висновку.

Директор ТОВ «Системи управління інформаційною безпекою»



**Віктор ЛУЦЕНКО**  
ініціали, прізвище

## Код Звіту 162 – Розрахунковий лист

```

//=====
// Звіт 162 - Розрахунковий лист (EN)
//
//=====
var str_rcd:string;
var i,LIC,LIC1,Y:integer;
var sm_n,sm_u,sm_s:double;
var flg_cmb5,flg_cmb6:integer;
var str_vo,str_m1,ss,str_m,str_y:string;
var smnu:double;
var masi:array[1..1] of string;
var mas_save:array[1..256] of string;
var sml,lmn,fkol1,i2,z:integer;
var DTUP,DTUP1,datn,datk:TDateTime;
var StrPdrSql,str_grppd:string;
var LDTUP:TDateTime;
var str_pd:String;
var flg_Pacc,flg_Tacc,flg_SEAcc,flg_KtSAcc:integer;
var CdFrm,kolmon:integer;
var strPacc,strTacc,strSEAcc,strKtSAcc:string;
var pDescr,pMaxLen,pMaxLen1,pResLen,pLineLen:Integer;
// ===== Для обробки TreeView
var pRec_Ct:integer;
var mspd:array[1..1]of string;
var l:integer;
var str_pcd:string;
var RcdMrk,RcdMrkC:string;
var RcdUncMrk:string;
var Tmp_RcdMrk:string;
var flg_Alt,pSubPdr:integer;
var pFilePath :String;
//=====
var strSQL,strSLD :String;
var strMemo21,strTheme,strTxtLetter :String;
var pStrText:String;
var pDt :TDateTime;
var LNG:byte;

function EnUa (En: string; Ua :string;) :string;
begin
  if LNG = 1 then
    Result := En
  else
    Result := Ua;
end;

function SetLine( pLen :Integer ):String;
var pStr :String;

```

```

var p:Integer;
begin
  pStr := "";
  For p:= 0 To pLen - 1 do
    begin
      pStr := pStr + '!';
    end;
  Result := pStr;
end;

// Отримати максимальну довжину найменування виду оплати
function GetVoNmMaxLen():Integer;
var pRet:Integer;
begin
  pRet := 0;
  RL.First;
  WHILE NOT RL.EOF DO
    begin
      If Length(<RL."Spr_Nm">) > pRet then
        begin
          pRet := Length(<RL."Spr_Nm"> );
        end;
      RL.Next;
    end;
  Result := pRet;
end;

// Отримати максимальну довжину суми
function GetSmMaxLen():Integer;
var pRet:Integer;
begin
  pRet := 0;
  RL.First;
  WHILE NOT RL.EOF DO
    begin
      If Length(FormatFloat('#,##0.00',<RL."КпуRl_Sm">)) > pRet then
        begin
          pRet := Length(FormatFloat('#,##0.00',<RL."КпуRl_Sm">));
        end;
      RL.Next;
    end;
  Result := pRet;
end;

// Доповнити найменування ВО необхідною кількістю прогалін
function VoNmSpace( pNm :String; pMaxLen:Integer; ):String;
var pRet:String;
var t:Integer;
begin
  t := 0;
  pRet := pNm;

```

```

If Length( pNm ) <= pMaxLen then
  begin
    For t :=0 To pMaxLen - Length( pNm ) do
      begin
        pRet := pRet + ' ';
      end;
    end
  Else
    begin
      pRet := pRet + ' ';
    end;
  Result := pRet;
end;

// pLen1 - рядок
// pLen2 - максимальна довжина найменування ВО
function GetSpace( pLen1:Integer; pLen2:Integer ):String;
var pStr :String;
var p,i:Integer;
begin
  pStr := ""; // Приймає значення 18 прогалін
  p := 0;
  If pLen1 = pLen2 then
    begin
      pStr := '          ';
    end
  Else If pLen1 < pLen2 then
    begin
      p := pLen2 - pLen1;
      pStr := '          ';
      For i := 0 To p - 1 do
        begin
          pStr := pStr + ' ';
        end;
      end
    Else If pLen1 > pLen2 then
      begin
        p := pLen1 - pLen2;
        pStr := Copy('          ',1,Length('          ')-p);
      end;
    Result := pStr;
end;

// Доповнити суму необхідною кількістю прогалін зліва
function VoSmSpace( pNm :String;pMaxLen:Integer; ):String;
var pRet:String;
var t:Integer;
begin
  t := 0;
  pRet := "";
  If Length( pNm ) <= pMaxLen then

```

```

begin
  For t :=0 To pMaxLen - Length( pNm ) do
    begin
      pRet := pRet + ' ';
    end;
  end
Else
  begin
    pRet := pRet + ' ';
  end;
pRet := pRet + pNm;
Result := pRet;
end;

// Визначає кількість місяців між двома датами
function KolMonth(dat1 :TDATETIME; dat2 :TDATETIME) :Integer;
var m1,m2,y1,y2 :integer;
begin
  m1 := MonthOf(dat1);
  m2 := MonthOf(dat2);
  y1 := YearOf(dat1);
  y2 := YearOf(dat2);
  Result := (y2-y1)*12+(m2-m1);
end;

// Отримати код батьківського підрозділу для угруповання
function GetGrpCd( pPD:String; pStr:String ):String;
var tmpStr:String;
var ResStr:String;
begin
  ResStr:="";
  tmpStr:= pStr;
  while Pos( ',', tmpStr )<>0 do
    begin
      If Copy( tmpStr, 1, Pos( ',', tmpStr )-1 )=Copy( pPD, 1, Length( Copy( tmpStr, 1, Pos( ',',
tmpStr )-1 ) ) ) then
        begin
          ResStr:=Copy( tmpStr, 1, Pos( ',', tmpStr )-1 );
          // break; 05.01.2007 wolf
        end;
        Delete( tmpStr, 1, Pos( ',', tmpStr ) );
      end;
      If tmpStr=Copy( pPD, 1, Length( tmpStr ) ) then
        begin
          ResStr:=Copy( tmpStr, 1, Length( tmpStr ) );
          // break; 05.01.2007 wolf
        end;
        Delete( tmpStr, 1, Length( tmpStr ) );
      Result:= ResStr;
    end;
  end;
end;

```

```

function DT_STR(D:TDateTime):string; //(dd.mm.yyyy)
var dd,mm:string;
begin
  If Length(DayOf(D))=1 then
    begin
      dd:='0'+IntToStr(DayOf(D));
    end
  Else If Length(DayOf(D))=2 then
    begin
      dd:=IntToStr(DayOf(D));
    end;
  If Length(MonthOf(D))=1 then
    begin
      mm:='0'+IntToStr(MonthOf(D));
    end
  Else If Length(MonthOf(D))=2 then
    begin
      mm:=IntToStr(MonthOf(D));
    end;
  Result:=dd+'.'+mm+'.'+IntToStr(YearOf(D));
end;

```

```

Function NVL(T:Integer):Integer;
const ms=[74,34,48,49,50,95,79,80,81,116,29,83,78];
var i:integer;
begin
  Result:=0;
  For i:=0 to 12 do
    begin
      If ms[i]=T then
        begin
          Result:=1;
          break;
        end;
    end;
end;

```

```

procedure Button2OnClick(Sender: TfrxComponent);
begin
  StopCloseReport;
end;

```

```

procedure Button1OnClick(Sender: TfrxComponent);
var sMes: string;
begin
  // ZVN 04/11/11 (524196){ 2930 - позначки у картотеці студентів
  If ( BPRO.BPRO_CD_MD() <> 543 ) then
    GetdsMark(120)
  else
    GetdsMark(2930);
  // ZVN 04/11/11 (524196)}

```

```

Page3.Visible :=False;
FRMPARM.Visible :=False;
Page4.Visible:= CheckBox8.Checked and CheckBox10.Checked;
DTUP :=
EncodeDate(YearOf(IncMonth(datn,Combobox1.ItemIndex)),MonthOf(IncMonth(datn,Combobox
1.ItemIndex)),1);
DTUP1 :=
EncodeDate(YearOf(IncMonth(datn,Combobox2.ItemIndex)),MonthOf(IncMonth(datn,Combobox
2.ItemIndex)),1);
LDTUP :=
EncodeDate(YearOf(IncMonth(datn,Combobox2.ItemIndex)),MonthOf(IncMonth(datn,Combobox
2.ItemIndex)),DaysInMonth(YearOf(IncMonth(datn,Combobox2.ItemIndex)),MonthOf(IncMonth(
datn,Combobox2.ItemIndex))));
If CheckBox8.Checked then
begin
sMes:=CheckMailParams();
If sMes <> " then
begin
showmessage(sMes);
Page2.ModalResult := mrNone;
end;
end;
If CheckBox7.Checked and ((BPRO.FILE_EXISTS(Edit1.Text) = False) OR
(Trim(Edit1.Text)=")) then
begin
showmessage('Каталог не існує');
Page2.ModalResult := mrNone;
end
Else
begin
SetUserRepParam('PPATH',Edit1.Text);
end;
SetUserRepParam('DT1',IntToStr(ComboBox1.ItemIndex));
SetUserRepParam('DT2',IntToStr(ComboBox2.ItemIndex));
SetUserRepParam('DT',FloatToStr(DateEdit1.Date));
If CheckBox4.Checked then
begin
SetUserRepParam('CB4','1');
end
Else
begin
SetUserRepParam('CB4','0');
end;
If CheckBox1.Checked then
begin
SetUserRepParam('CB1','1');
end
Else
begin
SetUserRepParam('CB1','0');
end;

```

```
If CheckBox2.Checked then
  begin
    SetUserRepParam('CB2','1');
  end
Else
  begin
    SetUserRepParam('CB2','0');
  end;
SetUserRepParam('CMB5',IntToStr(ComboBox5.ItemIndex));
SetUserRepParam('CMB6',IntToStr(ComboBox6.ItemIndex));
If CheckBox3.Checked then
  begin
    SetUserRepParam('CB3','1');
  end
Else
  begin
    SetUserRepParam('CB3','0');
  end;
If CheckBox6.Checked then
  begin
    SetUserRepParam('CB6','1');
  end
Else
  begin
    SetUserRepParam('CB6','0');
  end;
If CheckBox9.Checked then
  begin
    SetUserRepParam('CB9','1');
  end
Else
  begin
    SetUserRepParam('CB9','0');
  end;
If CheckBox7.Checked then
  begin
    SetUserRepParam('CB7','1');
  end
Else
  begin
    SetUserRepParam('CB7','0');
  end;
If CheckBox8.Checked then
  begin
    SetUserRepParam('CB8','1');
  end
Else
  begin
    SetUserRepParam('CB8','0');
  end;
If CheckBox10.Checked then
```

```

begin
  SetUserRepParam('CB10','1');
end
Else
begin
  SetUserRepParam('CB10','0');
end;
//ZVN 06/09/10
If CheckBox8_.Checked then
begin
  SetUserRepParam('PAR8','1');
end
Else
begin
  SetUserRepParam('PAR8','0');
end;
If CheckBox9.Checked then
begin
  SetUserRepParam('PAR4','1');
end
Else
begin
  SetUserRepParam('PAR4','0');
end;
If CheckBox9_.Checked then
begin
  SetUserRepParam('PAR9','1');
end
Else
begin
  SetUserRepParam('PAR9','0');
end;
end;

procedure Page2OnActivate(Sender: TfrxComponent);
begin
  If GetUserRepParam('CB5') = '1' then
begin
  pSubPdr := 1;
end
Else
begin
  pSubPdr := 0;
end;
If GetUserRepParam('CB4') = '1' then
begin
  CheckBox4.Checked := True;
end
Else
begin
  CheckBox4.Checked := False;

```

```
end;
If GetUserRepParam('CB1') = '1' then
begin
  CheckBox1.Checked := True;
end
Else
begin
  CheckBox1.Checked := False;
end;
If GetUserRepParam('CB2') = '1' then
begin
  CheckBox2.Checked := True;
end
Else
begin
  CheckBox2.Checked := False;
end;
If GetUserRepParam('CB6') = '0' then
begin
  CheckBox6.Checked := False;
end
Else
begin
  CheckBox6.Checked := True;
end;
If GetUserRepParam('CB9') = '1' then
begin
  CheckBox9.Checked := True;
end
Else
begin
  CheckBox9.Checked := False;
end;
If GetUserRepParam('CB7') = '1' then
begin
  CheckBox7.Checked := True;
end
Else
begin
  CheckBox7.Checked := False;
end;
If GetUserRepParam('CB8') = '1' then
begin
  CheckBox8.Checked := True;
end
Else
begin
  CheckBox8.Checked := False;
end;
If GetUserRepParam('CB10') = '1' then
begin
```

```

    CheckBox10.Checked := True;
end
Else
begin
    CheckBox10.Checked := False;
end;
If GetUserRepParam('PAR8') = '0' then
begin
    CheckBox8_.Checked := False;
end
Else
begin
    CheckBox8_.Checked := True;
end;
If GetUserRepParam('PAR9') = '1' then
begin
    CheckBox9_.Checked := True;
end
Else
begin
    CheckBox9_.Checked := False;
end;
If GetUserRepParam('PAR4') = '1' then
begin
    CheckBox9.Checked := True;
end
Else
begin
    CheckBox9.Checked := False;
end;
If ( BPRO.BPRO_CD_MD() = 84 ) OR ( BPRO.BPRO_CD_MD() = 101 ) OR (
BPRO.BPRO_CD_MD() = 576 ) OR ( BPRO.BPRO_CD_MD() = 90 ) then
begin
    Button3.Enabled := True;
    Button3.Visible := True;
end
Else
begin
    Button3.Enabled := False;
    Button3.Visible := False;
    str_pd:="//ZVN 01/11/11
end;
If GetUserRepParam('PPATH') = " then
begin
    Edit1.Text := 'C:\';
end
Else
begin
    Edit1.Text := GetUserRepParam('PPATH');
end;
CheckBox7OnClick(CheckBox7);

```

```

If CheckBox4.Checked=False then
  begin
    DateEdit1.Enabled:=False;
    ComboBox1.Enabled:=True;
    ComboBox2.Enabled:=True;
  end
Else
  begin
    DateEdit1.Enabled:=True;
    ComboBox1.Enabled:=False;
    ComboBox2.Enabled:=False;
  end;
qDat.Open;
datn := <qDat."datn">;
If (GetCargoVal('BpI_CdMd') = 576) then
  begin
    If GetUserRepParam('DT') <> " then
      begin
        DateEdit1.Date := StrToFloat(GetUserRepParam('DT'));
      end
    Else
      begin
        DateEdit1.Date:=GetCargoVal('sPayrep_DatUp');
      end;
    datk := GetCargoVal('sPayRep_DatUp');
    // VDG 23/02/2009 СОИД 287232 {
    //ComboBox1.Enabled := False;
    //ComboBox2.Enabled := False;
    //CheckBox4.Enabled := False;
    //DateEdit1.Enabled := False;
    // VDG 23/02/2009 СОИД 287232 }
    Button3.Enabled := False;
    str_pd := "";
  end
Else
  begin
    If GetUserRepParam('DT') <> " then
      begin
        DateEdit1.Date := StrToFloat(GetUserRepParam('DT'));
      end
    Else
      begin
        DateEdit1.Date:=BPRO.CUR_UP().Value;
      end;
    datk := BPRO.CUR_UP().Value;
  end;
kolmon := KolMonth(datn,datk);
datk := datn;
For i := 0 to kolmon do
  begin

```

```

If datk = BPRO.CUR_UP().Value then
begin
  Combobox1.Items.add(' '+BPRO.FMONTH_NAME(datk,1)+'
'+chr(39)+Copy(inttostr(YearOf(datk)),3,2)+ENUA(' (current)', ' (поточний)'));
  Combobox2.Items.add(' '+BPRO.FMONTH_NAME(datk,1)+'
'+chr(39)+Copy(inttostr(YearOf(datk)),3,2)+ENUA(' (current)', ' (поточний)'));
end
Else
begin
  Combobox1.Items.add(' '+BPRO.FMONTH_NAME(datk,1)+'
'+chr(39)+Copy(inttostr(YearOf(datk)),3,2)+ENUA(' (closed)', ' (закритий)'));
  Combobox2.Items.add(' '+BPRO.FMONTH_NAME(datk,1)+'
'+chr(39)+Copy(inttostr(YearOf(datk)),3,2)+ENUA(' (closed)', ' (закритий)'));
end;
datk := IncMonth(datk,1);
end;
If GetUserRepParam('DT1') = " then
begin
  Combobox1.ItemIndex := kolmon;
end
Else
begin
  If StrToInt(GetUserRepParam('DT1')) > kolmon then
begin
  Combobox1.ItemIndex := kolmon;
end
Else
begin
  If GetCargoVal('BpI_CdMd') <> 576 then
begin
  Combobox1.ItemIndex := StrToInt(GetUserRepParam('DT1'));
end
Else
begin
  Combobox1.ItemIndex := kolmon;
end;
end;
end;
If GetUserRepParam('DT2') = " then
begin
  Combobox2.ItemIndex := kolmon;
end
Else
begin
  If StrToInt(GetUserRepParam('DT2')) > kolmon then
begin
  Combobox2.ItemIndex := kolmon;
end
Else
begin
  If GetCargoVal('BpI_CdMd') <> 576 then

```

```

begin
  Combobox2.ItemIndex := StrToInt(GetUserRepParam('DT2'));
end
Else
begin
  Combobox2.ItemIndex := kolmon;
end;
end;
end;
masi[1]:= "";
For i:=1 to 128 do
begin
  If GetUserRepParam(mas_save[i])<>" then
  begin
    masi[1]:=masi[1]+GetUserRepParam(mas_save[i]);
  end;
end;
str_vo:= "";
For i:=129 to 256 do
begin
  If GetUserRepParam(mas_save[i])<>" then
  begin
    str_vo:=str_vo+GetUserRepParam(mas_save[i]);
  end;
end;
VO.First;
while not VO.Eof do
begin
  CheckListBox1.Items.AddObject(<VO."VO_CDCHR">+'
'+<VO."SPR_NM">,TObject(<VO."VO_CD">));
  VO.Next;
end;
fkol1:=CheckListBox1.Items.Count-1;
For i:=0 to CheckListBox1.Items.Count-1 do
begin
  CheckListBox1.Checked[i]:=False;
end;
If masi[1]<>" then
begin
  while Pos(',',masi[1])>0 do
  begin
    If fkol1>=StrToInt(Copy(masi[1],1,Pos(',',masi[1])-1)) then
      CheckListBox1.Checked[StrToInt(Copy(masi[1],1,Pos(',',masi[1])-1))]:=True;
    delete(masi[1], 1, pos(',',masi[1]));
  end;
  If fkol1>=StrToInt(masi[1]) then
    CheckListBox1.Checked[StrToInt(masi[1])]:=True;
end
Else
begin
  If fkol1>0 then

```

```

    CheckListBox1.Checked[0]:=True;
end;
For i:=0 to CheckListBox1.Items.Count-1 do
begin
    If CheckListBox1.Checked[i] then
        begin
            str_vo:=str_vo+varostr(CheckListBox1.Items.Objects[i])+',';
        end;
    end;
If GetUserRepParam('CB3') = '1' then
begin
    CheckBox3.Checked := True;
end
Else
begin
    CheckBox3.Checked := False;
end;
If (CheckBox3.Checked=False) and (str_vo=") then
begin
    CheckListBox1.Enabled:=False;
end
Else
begin
    CheckListBox1.Enabled:=True;
end;
If CheckListBox1.Items.Count <> 0 then
begin
    CheckBox3.Checked:=True;
    If CheckBox3.Checked=False then
        begin
            CheckListBox1.Enabled:=False;
        end
    Else
        begin
            CheckListBox1.Enabled:=True;
        end;
end
Else
begin
    CheckBox3.Checked:=False;
end;
// Джерела фінансування
SF.First;
If SF.Eof then
    ComboBox5.Enabled:=False;
while not SF.Eof do
begin
    ComboBox5.Items.AddObject(<SF."PAYSF_CD">+'
'+<SF."PAYSF_NM">,TObject(<SF."PAYSF_RCD">));
    SF.Next;
end;

```

```

// Економічні класифікатори витрат
EKR.First;
If EKR.Eof then
  ComboBox6.Enabled:=False;
  while not EKR.Eof do
    begin
      ComboBox6.Items.AddObject(<EKR."PAYEK_CD">+'
'+<EKR."PAYEK_NM">,TObject(<EKR."PAYEK_RCD">));
      EKR.Next;
    end;
  If ComboBox5.ItemIndex=-1 then
    ComboBox5.Text:=ENUA('Все','Всі');
  If ComboBox6.ItemIndex=-1 then
    ComboBox6.Text:=ENUA('Все','Всі');
  end;

procedure Page1OnAfterPrint(Sender: TfrxComponent);
begin
  // Очищення ServerMrk
  BPRO.DEL_MRK_SRV(120);
  BPRO.DEL_MRK_SRV(2930);// ZVN 04/11/11 (524196)
end;

procedure GroupHeader2OnBeforePrint(Sender: TfrxComponent);
begin
  pDt:=0;
  // Пільги податку
  BPRO.LS_BY_TN(<Query1."Кпу_Tn">);
  //BPRO.LS_GET_LAST(<Query1."КпуRl_DatUp">); // VDG 10/06/2008 СОИД
253080
  BPRO.LS_GET_LAST_UP(<Query1."КпуRl_DatUp">,<Query1."КпуRl_DatUp">); //
VDG 10/06/2008 СОИД 253080
  BPRO.LS_GET_NM();
  str_y := IntToStr(<Query1."Y">);
  If <Query1."M">=1 then
    begin
      str_m:=ENUA('january','січень');
      str_m1:=ENUA('february','лютий');
    end
  Else If <Query1."M">=2 then
    begin
      str_m:=ENUA('february','лютий');
      str_m1:=ENUA('march','березень');
    end
  Else If <Query1."M">=3 then
    begin
      str_m:=ENUA('march','березень');
      str_m1:=ENUA('april','квітень');
    end
  Else If <Query1."M">=4 then
    begin

```

```

    str_m:=ENUA('april','квітень');
    str_m1:=ENUA('may','травень');
end
Else If <Query1."M">=5 then
begin
    str_m:=ENUA('may','травень');
    str_m1:=ENUA('june','червень');
end
Else If <Query1."M">=6 then
begin
    str_m:=ENUA('june','червень');
    str_m1:=ENUA('july','липень');
end
Else If <Query1."M">=7 then
begin
    str_m:=ENUA('july','липень');
    str_m1:=ENUA('august','серпень');
end
Else If <Query1."M">=8 then
begin
    str_m:=ENUA('august','серпень');
    str_m1:=ENUA('september','вересень');
end
Else If <Query1."M">=9 then
begin
    str_m:=ENUA('september','вересень');
    str_m1:=ENUA('october','жовтень');
end
Else If <Query1."M">=10 then
begin
    str_m:=ENUA('october','жовтень');
    str_m1:=ENUA('november','листопад');
end
Else If <Query1."M">=11 then
begin
    str_m:=ENUA('november','листопад');
    str_m1:=ENUA('december','грудень');
end
Else If <Query1."M">=12 then
begin
    str_m:=ENUA('december','грудень');
    str_m1:=ENUA('january','січень');
    str_y := IntToStr(<Query1."Y">+1);
end;
If CheckBox4.Checked=False then
begin
    Memo21.Text:=Uppercase(str_m)+' '+Uppercase(IntToStr(<Query1."Y">));
    SLD.Close;
    If CheckBox9.Checked then
begin
    if (GetServerType()='MSSQL') and (<Query1."KPU_TN"> > 2147483647)

```

```

then SLD.ParamByname('TN2').DataType := ftLargeInt
else SLD.ParamByname('TN2').DataType := ftInteger;
  SLD.ParamByName('TN2').Value:=<Query1."KPU_TN">;
end;
if (GetServerType()='MSSQL') and (<Query1."KPU_TN"> > 2147483647)
then SLD.ParamByname('TN1').DataType := ftLargeInt
else SLD.ParamByname('TN1').DataType := ftInteger;
SLD.ParamByName('TN1').Value:=<Query1."KPU_TN">;
SLD.ParamByName('M').DataType:=ftInteger;
SLD.ParamByName('M').Value:=<Query1."M">;
SLD.ParamByName('Y').DataType:=ftInteger;
SLD.ParamByName('Y').Value:=<Query1."Y">;
SLD.Open;
If SLD.Eof then
  begin
    Memo28.Visible:=False;
    Memo29.Visible:=False;
  end
Else
  begin
    Memo28.Visible:=True;
    Memo29.Visible:=True;
    Memo28.Text:='БОПГ HA '+UpperCase(str_m)+'
'+UpperCase(IntToStr(<Query1."Y">));
  end;
end
Else
  begin
    Memo21.Text:=DT_STR(DateEdit1.Date);
    Memo28.Text:=";
    Memo29.Text:=";
  end;
sm_n:=0;
sm_s:=0;
sm_u:=0;
RL.Close;
If CheckBox4.Checked=False then
  begin
    RL.ParamByName('rcd').DataType:=ftInteger;
    RL.ParamByName('rcd').Value:=<Query1."KPU_RCD">;
    RL.ParamByName('m').DataType:=ftInteger;
    RL.ParamByName('y').DataType:=ftInteger;
    RL.ParamByName('m').Value:=<Query1."M">;
    RL.ParamByName('y').Value:=<Query1."Y">;
  end
Else
  begin
    RL.ParamByName('rcd').DataType:=ftInteger;
    RL.ParamByName('rcd').Value:=<Query1."KPU_RCD">;
    RL.ParamByName('DT').DataType:=ftDate;
    RL.ParamByName('DT').Value:=DateEdit1.Date;
  end

```

```

end;
RL.First;
// VDG 27/03/2008 СОИД 235215 ===== {
If CheckBox6.Checked then
begin
  If RL.EOF then
  begin
    MasterData1.Visible := false;
    DetailData1.Visible := false;
    GroupHeader1.Visible := false;
    GroupHeader2.Visible := false;
    GroupHeader3.Visible := false;
    GroupFooter1.Visible := false;
    GroupFooter2.Visible := false;
    GroupFooter3.Visible := false;
    If BPRO.BPRO_CD_MD() = 520 then
    begin
      showmessage('Звіт порожній');
      StopCloseReport;
    end;
  end
Else
begin
  MasterData1.Visible := True;
  DetailData1.Visible := True;
  GroupHeader1.Visible := True;
  GroupHeader2.Visible := True;
  GroupHeader3.Visible := True;
  GroupFooter1.Visible := True;
  GroupFooter2.Visible := True;
  GroupFooter3.Visible := True;
end;
end
Else
begin
  MasterData1.Visible := True;
  DetailData1.Visible := True;
  GroupHeader1.Visible := True;
  GroupHeader2.Visible := True;
  GroupHeader3.Visible := True;
  GroupFooter1.Visible := True;
  GroupFooter2.Visible := True;
  GroupFooter3.Visible := True;
end;
// VDG 27/03/2008 СОИД 235215 ===== }
// Експорт розрахункових листів
// Створення файлу, запис шапки
If CheckBox7.Checked then
begin
  strTxtLetter := "";
  strMemo21 := Memo21.Text;

```

```

// Для працівників, у яких вказано EMail
//If <Query1."Кпу_EMail"> <> " then
begin
  pFilePath:=Edit1.Text+'\'+VarToStr(<Query1."Кпу_Tn">);
  If <Query1."M"> < 10 then
    pFilePath:=pFilePath+'_0';
  pFilePath:=pFilePath+Copy(IntToStr(<Query1."Y">),3,2)+'.html';
  pDescr := BPRO.FILE_OPEN(pFilePath,1);
  BPRO.FILE_WRITE(pDescr,'<html>');
  strTxtLetter := strTxtLetter + '<html>'+Chr(13);
  BPRO.FILE_WRITE(pDescr,'<head>');
  strTxtLetter := strTxtLetter + '<head>'+Chr(13);
  BPRO.FILE_WRITE(pDescr,'<meta http-equiv="Content-Language"
content="En">');
  strTxtLetter := strTxtLetter + '<meta http-equiv="Content-Language"
content="En">'+Chr(13);
  strTxtLetter := strTxtLetter + '<meta name="GENERATOR" content="Microsoft
FrontPage 6.0">'+Chr(13);
  BPRO.FILE_WRITE(pDescr,'<meta name="GENERATOR" content="Microsoft
FrontPage 6.0">');
  strTxtLetter := strTxtLetter + '<meta name="ProgId"
content="FrontPage.Editor.Document">'+Chr(13);
  BPRO.FILE_WRITE(pDescr,'<meta name="ProgId"
content="FrontPage.Editor.Document">');
  BPRO.FILE_WRITE(pDescr,'<meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">');
  strTxtLetter := strTxtLetter + '<meta http-equiv="Content-Type" content="text/html;
charset=windows-1251">'+Chr(13);
  strTxtLetter := strTxtLetter + '</head>'+Chr(13);
  BPRO.FILE_WRITE(pDescr,'</head>');
  BPRO.FILE_WRITE(pDescr,'<body bgcolor="#FFFFFF" text="#000000">');
  strTxtLetter := strTxtLetter + '<body bgcolor="#FFFFFF" text="#000000">'+Chr(13);
  strTxtLetter := strTxtLetter + '<h3>'+ENUA('SETTLEMENT SHEET FOR
','ПОЗРАХУНКОВИЙ ЛИСТ ЗА')+TRIM(strMemo21)+'</h3>'+Chr(13);
  BPRO.FILE_WRITE(pDescr,'<title>');
  BPRO.FILE_WRITE(pDescr,ENUA('SETTLEMENT SHEET FOR
','ПОЗРАХУНКОВИЙ ЛИСТ ЗА')+TRIM(strMemo21));
  BPRO.FILE_WRITE(pDescr,'</title>');
  BPRO.FILE_WRITE(pDescr,'<h3>'+ENUA('SETTLEMENT SHEET FOR
','ПОЗРАХУНКОВИЙ ЛИСТ ЗА')+TRIM(strMemo21)+'</h3>');
  strTxtLetter := strTxtLetter + '<pre>' + Chr(13);
  BPRO.FILE_WRITE(pDescr,'<pre>');
  BPRO.FILE_WRITE(pDescr,'<style='+Chr(39)+'line-height:15pt'+Chr(39)+'>');
  strTxtLetter := strTxtLetter + '<style='+Chr(39)+'line-height:15pt'+Chr(39)+'>' +
Chr(13);
  BPRO.FILE_WRITE(pDescr,'<font size=3>'+ENUA('Registration number...:
','Табельний номер...:')+IntToStr(<Query1."Кпу_Tn">)+'</font>');
  strTxtLetter := strTxtLetter + '<font size=3>'+ENUA('Registration number...:
','Табельний номер...:')+IntToStr(<Query1."Кпу_Tn">)+'</font>' + Chr(13);
  BPRO.FILE_WRITE(pDescr,'<font size=3>'+ENUA('Ф.И.О.....: ','П.І.Б.....:
')+TRIM(<Query1."Кпу_Fio">)+'</font>');

```

```

strTxtLetter := strTxtLetter + '<font size=3>'+ENUA('Ф.И.О.....: ',П.І.Б.....:
')+TRIM(<Query1."Kpu_Fio">)+</font>' + Chr(13);
BPRO.FILE_WRITE(pDescr,<font size=3>'+ENUA('Position.....:
','Посада.....: ') + TRIM(<Query1."SprD_NmFull">+<Query1."Sp_Nm">)+</font>');
strTxtLetter := strTxtLetter + '<font size=3>'+ENUA('Position.....:
','Посада.....: ') + TRIM(<Query1."SprD_NmFull">+<Query1."Sp_Nm">)+</font>' + Chr(13);
BPRO.FILE_WRITE(pDescr,<font size=3>'+ENUA('Subdivision.....:
','Підрозділ.....: ') + TRIM(<Query1."SprPdr_NmFull">)+</font>');
strTxtLetter := strTxtLetter + '<font size=3>'+ENUA('Subdivision.....:
','Підрозділ.....: ') + TRIM(<Query1."SprPdr_NmFull">)+</font>' + Chr(13);
BPRO.FILE_WRITE(pDescr,<font size=3>'+Оклад.....:
'+FormatFloat('#,##0.00',SETMT(<Query1."kpuprkz_okl">,<Query1."kpuprkz_kfcm">))+</font>
');
strTxtLetter := strTxtLetter + '<font size=3>'+Оклад.....:
'+FormatFloat('#,##0.00',SETMT(<Query1."kpuprkz_okl">,<Query1."kpuprkz_kfcm">))+</font>
+ Chr(13);
BPRO.FILE_WRITE(pDescr,<font size=3>'+ENUA('Tax benefits.....: ',Пільги
податку.....: ') + BPRO.LS_LG_NL_NM.Value+</font>');
strTxtLetter := strTxtLetter + '<font size=3>'+ENUA('Tax benefits.....: ',Пільги
податку.....: ') + BPRO.LS_LG_NL_NM.Value+</font>' + Chr(13);
BPRO.FILE_WRITE(pDescr,<font size=3>'+ENUA('Identificational №:
','Ідентифікаційний №: ') + TRIM(<Query1."Kpu_CdNlp">)+</font>');
strTxtLetter := strTxtLetter + '<font size=3>'+ENUA('Identificational №:
','Ідентифікаційний №: ') + TRIM(<Query1."Kpu_CdNlp">)+</font>' + Chr(13);
BPRO.FILE_WRITE(pDescr,</pre>');
strTxtLetter := strTxtLetter + '</pre>' + Chr(13);
pMaxLen := GetVoNmMaxLen();
pMaxLen1 := GetSmMaxLen();
If <SLD."kpurl_sin"> >= 0 then
begin
BPRO.FILE_WRITE(pDescr,<hr></hr>');
strTxtLetter := strTxtLetter + '<hr></hr>' + Chr(13);
BPRO.FILE_WRITENC(pDescr,<font size=4 face="Arial">'+ENUA('THE
ENTERPRISE'S DEBT','БОРГ ПІДПРИЄМСТВА НА ') + UpperCase(str_m)+
'+UpperCase(IntToStr(<Query1."Y">))+':'+FormatFloat('#,##0.00',SETMT(<SLD."kpurl_sin">,<M
T."SVL_ACC">))+</font>');
strTxtLetter := strTxtLetter + '<font size=4 face="Arial">'+ENUA('THE
ENTERPRISE'S DEBT','БОРГ ПІДПРИЄМСТВА НА ') + UpperCase(str_m)+
'+UpperCase(IntToStr(<Query1."Y">))+':'+FormatFloat('#,##0.00',SETMT(<SLD."kpurl_sin">,<M
T."SVL_ACC">))+</font>' + Chr(13);
BPRO.FILE_WRITE(pDescr,<hr></hr>');
strTxtLetter := strTxtLetter + '<hr></hr>' + Chr(13);
end
Else If <SLD."kpurl_sin"> < 0 then
begin
BPRO.FILE_WRITE(pDescr,<hr></hr>');
strTxtLetter := strTxtLetter + '<hr></hr>' + Chr(13);
BPRO.FILE_WRITENC(pDescr,<font size=4
face="Arial">'+ENUA('EMPLOYEE'S DEBT НА ', 'БОРГ ПРАЦІВНИКА НА
')+UpperCase(str_m)+

```

```

'+UpperCase(IntToStr(<Query1."Y">))+':'+FormatFloat('#,##0.00',SETMT(<SLD."kpurl_sin">,<M
T."SVL_ACC">))+</font>');
      strTxtLetter := strTxtLetter + '<font size=4 face="Arial">'+ENUA('EMPLOYEE'S
DEBT HA ','БОРГ ПРАЦІВНИКА НА ')+UpperCase(str_m)+'
'+UpperCase(IntToStr(<Query1."Y">))+':'+FormatFloat('#,##0.00',SETMT(<SLD."kpurl_sin">,<M
T."SVL_ACC">))+</font>' + Chr(13);
      end;
    end;
  end;
end;

procedure Memo17OnAfterData(Sender: TfrxComponent);
begin
  If Value=0 then
    Memo17.Text:="";
end;

procedure Memo18OnAfterData(Sender: TfrxComponent);
begin
  If Value=0 then
    Memo18.Text:="";
end;

procedure MasterData1OnBeforePrint(Sender: TfrxComponent);
begin
  If Checkbox7.Checked and (pDescr <> 0) then
    begin
      BPRO.FILE_WRITE(pDescr,'<table cellpadding=10>');
      strTxtLetter := strTxtLetter + '<table cellpadding=10>' + Chr(13);
    end;
  pDt:=0;
end;

procedure GroupHeader3OnBeforePrint(Sender: TfrxComponent);
begin
  pDt:=0;
  If <RL."GRP">=0 then
    begin
      Memo22.Text:=ENUA('ACCRUAL FEES:', 'НАРАХУВАННЯ:');
      If (CheckBox7.Checked) then
        begin
          //BPRO.FILE_WRITE(pDescr,'<table cellpadding=10>');
          //strTxtLetter := strTxtLetter + '<table cellpadding=10>' + Chr(13);
        end;
    end
  Else If <RL."GRP">=1 then
    begin
      Memo22.Text:=ENUA('RESTRAINTS:', 'УТРИМАННЯ:');
      If (CheckBox7.Checked) then
        begin
          //BPRO.FILE_WRITE(pDescr,'<table cellpadding=10>');

```

```

        //strTxtLetter := strTxtLetter + '<table cellpadding=10>' + Chr(13);
    end;
end;
end;

procedure Memo23OnAfterData(Sender: TfrxComponent);
begin
    If <RL."GRP">=0 then
        begin
            Memo23.Text:=ENUA('TOTAL ACCRUED:','ВСЬОГО НАРАХОВАНО:');
            If (CheckBox7.Checked) then
                begin
                    //BPRO.FILE_WRITE(pDescr,'</table>');
                    //strTxtLetter := strTxtLetter + '</table>' + Chr(13);
                    //BPRO.FILE_WRITE(pDescr,'<hr></hr>');
                    //strTxtLetter := strTxtLetter + '<hr></hr>' + Chr(13);
                    //BPRO.FILE_WRITENC(pDescr,'<font size=4 face="Arial">'+ВСЬОГО
НАРАХОВАНО:'+'</font>');
                    //strTxtLetter := strTxtLetter + '<font size=4 face="Arial">'+ВСЬОГО
НАРАХОВАНО:'+'</font>';
                    //BPRO.FILE_WRITE(pDescr,'<font size=4
face="Arial">'+FormatFloat('#,##0.00',smnu)+'</font>');
                    //strTxtLetter := strTxtLetter + '<font size=4
face="Arial">'+FormatFloat('#,##0.00',smnu)+'</font>' + Chr(13);
                    //BPRO.FILE_WRITE(pDescr,'<hr></hr>');
                    //strTxtLetter := strTxtLetter + '<hr></hr>' + Chr(13);
                    BPRO.FILE_WRITENC(pDescr,'<TR><TH ALIGN=LEFT
COLSPAN=5>'+ENUA('TOTAL ACCRUED:','ВСЬОГО НАРАХОВАНО:')+'</TH>');
                    strTxtLetter := strTxtLetter + '<TR><TH ALIGN=LEFT
COLSPAN=5>'+ENUA('TOTAL ACCRUED:','ВСЬОГО НАРАХОВАНО:')+'</TH>';
                    BPRO.FILE_WRITE(pDescr,'<TH
ALIGN=RIGHT>'+FormatFloat('#,##0.00',smnu)+'</TH></TR>');
                    strTxtLetter := strTxtLetter + '<TH
ALIGN=RIGHT>'+FormatFloat('#,##0.00',smnu)+'</TH></TR>' + Chr(13);
                end;
            end
        Else If <RL."GRP">=1 then
            begin
                Memo23.Text:=ENUA('TOTAL HOLDED:','ВСЬОГО УТРИМАНО:');
                If (CheckBox7.Checked) then
                    begin
                        //BPRO.FILE_WRITE(pDescr,'</table>');
                        //strTxtLetter := strTxtLetter + '</table>' + Chr(13);
                        //BPRO.FILE_WRITE(pDescr,'<hr></hr>');
                        //strTxtLetter := strTxtLetter + '<hr></hr>' + Chr(13);
                        //BPRO.FILE_WRITENC(pDescr,'<font size=4 face="Arial">'+ВСЬОГО
УТРИМАНО:'+'</font>');
                        //strTxtLetter := strTxtLetter + '<font size=4 face="Arial">'+ВСЬОГО
УТРИМАНО:'+'</font>';
                        //BPRO.FILE_WRITE(pDescr,'<font size=4
face="Arial">'+FormatFloat('#,##0.00',smnu)+'</font>');

```

```

        //strTxtLetter := strTxtLetter + '<font size=4
face="Arial">'+FormatFloat('#,##0.00',smnu)+'</font>' + Chr(13);
        //BPRO.FILE_WRITE(pDescr,'<hr></hr>');
        //strTxtLetter := strTxtLetter + '<hr></hr>' + Chr(13);
        BPRO.FILE_WRITENC(pDescr,'<TR><TH ALIGN=LEFT
COLSPAN=5>'+ENUA('TOTAL HOLDED:','ВСЬОГО УТРИМАНО:')+'</TH>');
        strTxtLetter := strTxtLetter + '<TR><TH ALIGN=LEFT
COLSPAN=5>'+ENUA('TOTAL HOLDED:','ВСЬОГО УТРИМАНО:')+'</TH>';
        BPRO.FILE_WRITE(pDescr,'<TH
ALIGN=RIGHT>'+FormatFloat('#,##0.00',smnu)+'</TH></TR>');
        strTxtLetter := strTxtLetter + '<TH
ALIGN=RIGHT>'+FormatFloat('#,##0.00',smnu)+'</TH></TR>' + Chr(13);
    end;
end;
end;

procedure Memo24OnAfterData(Sender: TfrxComponent);
begin
    If <RL."GRP">=0 then
        begin
            sm_n:=Value;
        end
    Else IF <RL."GRP">=1 then
        begin
            sm_u:=Value;
        end;
end;

procedure Memo25OnAfterData(Sender: TfrxComponent);
begin
    Memo25.Text:=ENUA('DEBT ON ','БОРГ НА')+UpperCase(str_m1)+'
'+UpperCase(str_y);
    If CheckBox4.Checked=True then
        Memo25.Text:="";
end;

procedure CheckBox3OnClick(Sender: TfrxComponent);
begin
    If CheckBox3.Checked=False then
        begin
            CheckListBox1.Enabled:=False;
        end
    Else
        begin
            CheckListBox1.Enabled:=True;
        end;
    If (str_vo="") and (CheckBox3.Checked=True) then
        begin
            Button1.Enabled:=False;
        end
    Else

```

```

begin
  Button1.Enabled:=True;
end;
end;

procedure ComboBox5OnChange(Sender: TfrxComponent);
var t:integer;
var s:string;
begin
  If flg_cmb5=1 then
    begin
      t:=ComboBox5.ItemIndex;
      s:=ComboBox5.Text;
    end;
  If Ord(Copy(ComboBox5.Text,1,1))=32 then
    begin
      ComboBox5.ItemIndex:=-1;
      ComboBox5.Text:=ENUA('All','Bci');
    end
  Else IF ComboBox5.Text="" then
    begin
      ComboBox5.ItemIndex:=-1;
      ComboBox5.Text:=ENUA('All','Bci');
    end
  Else If (Ord(Copy(ComboBox5.Text,1,1))<>32) then
    begin
      ComboBox5.ItemIndex:=t;
      ComboBox5.Text:=s;
    end;
  flg_cmb5:=0;
end;

procedure ComboBox6OnChange(Sender: TfrxComponent);
var f:integer;
var ss:string;
begin
  IF flg_cmb6=1 then
    begin
      f:=ComboBox6.ItemIndex;
      ss:=ComboBox6.Text;
    end;
  If Ord(Copy(ComboBox6.Text,1,1))=32 then
    begin
      ComboBox6.ItemIndex:=-1;
      ComboBox6.Text:=ENUA('All','Bci');
    end
  Else If ComboBox6.Text="" then
    begin
      ComboBox6.ItemIndex:=-1;
      ComboBox6.Text:=ENUA('All','Bci');
    end
end

```

```

strSQL := strSQL +
'payvo1.vo_cdchr,payvo1.vo_cdchrs,kpurlo1.kpurl_Sm,pspr.spr_nm,' + Chr(13);
//23/11/12 ZVN (578850){strSQL := strSQL +
'kpux.kpu_rcd,kpurlo1.kpurl_days,kpurlo1.kpurl_hrs,' + Chr(13);
strSQL := strSQL + 'kpux.kpu_rcd,(case when BIT_AND( KpuRl1.KpuRl_Prz , 8 )
= 0 then KpuRl1.kpurl_days else -KpuRl1.kpurl_days end) as kpurl_days,'+Chr(13);
strSQL := strSQL + '(case when BIT_AND( KpuRl1.KpuRl_Prz , 8 ) = 0 then
KpuRl1.kpurl_hrs else -KpuRl1.kpurl_hrs end) as kpurl_hrs,' + Chr(13);
//23/11/12 ZVN (578850)}
strSQL := strSQL +
'kpurlo1.kpurl_datup,payvo1.vo_grp,payvo1.vo_nur,kpurlo1.kpurlSvm_tn,' + Chr(13);
strSQL := strSQL +
'kpurlo1.kpurl_sf,kpurlo1.kpurl_ek,kpurlo1.kpurl_datrp,kpurlo1.kpurl_cdvo,BIT_AND(
kpurlo1.kpurl_prz , 65536 ) as svm,payvo1.vo_met' + Chr(13);
strSQL := strSQL + 'from kpurlo1' + Chr(13);
strSQL := strSQL + 'inner join payvo1 on payvo1.vo_cd=kpurlo1.kpurl_cdvo' +
Chr(13);
strSQL := strSQL + 'inner join pspr on payvo1.vo_cd=pspr.spr_cd and
pspr.sprspr_cd=787202' + Chr(13);
strSQL := strSQL + 'inner join kpux on kpurlo1.kpu_tn=kpux.kpu_tn' + Chr(13);
strSQL := strSQL + 'where payvo1.vo_grp<=127' + Chr(13);
// VDG 29/05/2009 СОИД 282024 {
If CheckBox9.Checked = False then
begin
strSQL := strSQL + 'AND BIT_AND( kpurlo1.kpurl_prz , 65536 ) = 0' + Chr(13);
end;
// VDG 29/05/2009 СОИД 282024 }
// Вибірково з ІФ
If ComboBox5.ItemIndex<>-1 then
begin
strSQL := strSQL + 'and
kpurl_sf='+vartostr(ComboBox5.Items.Objects[ComboBox5.ItemIndex])+"' + Chr(13);
end;
// вибірково за ЕКР
If ComboBox6.ItemIndex<>-1 then
begin
strSQL := strSQL + 'and
kpurl_ek='+vartostr(ComboBox6.Items.Objects[ComboBox6.ItemIndex])+"' + Chr(13);
end;
strSQL := strSQL + 'union all' + Chr(13);
strSQL := strSQL + 'select Month(kpurlo1.kpurl_datup) as M_UP,' + Chr(13);
strSQL := strSQL + 'Year(kpurlo1.kpurl_datup) as Y_UP,' + Chr(13);
strSQL := strSQL + 'payvo1.vo_cdchr,payvo1.vo_cdchrs,sum(kpurlo1.kpurl_Sm) as
kpurl_Sm,pspr.spr_nm,' + Chr(13);
strSQL := strSQL + 'kpux.kpu_rcd,SUM(case when BIT_AND( KpuRl_Prz , 8 ) = 0
then kpurl_days else -kpurl_days end) as kpurl_days,SUM(case when BIT_AND( KpuRl_Prz , 8 )
= 0 then kpurl_hrs else -kpurl_hrs end) as kpurl_hrs,' + Chr(13);
strSQL := strSQL + 'kpurlo1.kpurl_datup,payvo1.vo_grp,payvo1.vo_nur,0 AS
kpurlSvm_tn,' + Chr(13);
strSQL := strSQL + '0 AS kpurl_sf,0 AS
kpurl_ek,kpurlo1.kpurl_datrp,kpurlo1.kpurl_cdvo,0 AS svm,payvo1.vo_met' + Chr(13);

```

```

strSQL := strSQL + 'from kpurlo1' + Chr(13);
strSQL := strSQL + 'inner join payvo1 on payvo1.vo_cd=kpurlo1.kpurl_cdvo' +
Chr(13);
strSQL := strSQL + 'inner join pspr on payvo1.vo_cd=pspr.spr_cd and
pspr.sprspr_cd=787202' + Chr(13);
strSQL := strSQL + 'inner join kpux on kpurlo1.kpu_tn=kpux.kpu_tn' + Chr(13);
strSQL := strSQL + 'where payvo1.vo_grp>127' + Chr(13);
//ZVN 06/09/10{
If CheckBox9.Checked=True then
strSQL := strSQL + ' and payvo1.vo_grp!=130' + Chr(13);
// VDG 29/05/2009 СОИД 282024 {
If CheckBox9.Checked = False then
begin
strSQL := strSQL + 'AND BIT_AND( kpurlo1.kpurl_prz , 65536 ) = 0' + Chr(13);
end;
// VDG 29/05/2009 СОИД 282024 }
// Вибірково з ІФ
If ComboBox5.ItemIndex<>-1 then
begin
strSQL := strSQL + 'and
kpurl_sf='+vartostr(ComboBox5.Items.Objects[ComboBox5.ItemIndex])+' + Chr(13);
end;
// вибірково за ЕКР
If ComboBox6.ItemIndex<>-1 then
begin
strSQL := strSQL + 'and
kpurl_ek='+vartostr(ComboBox6.Items.Objects[ComboBox6.ItemIndex])+' + Chr(13);
end;
strSQL := strSQL + 'group by
Month(kpurlo1.kpurl_datup),Year(kpurlo1.kpurl_datup),' + Chr(13);
strSQL := strSQL + 'payvo1.vo_cdchr,payvo1.vo_cdchrs,pspr.spr_nm,kpux.kpu_rcd,'
+ Chr(13);
strSQL := strSQL +
'kpurlo1.kpurl_datup,payvo1.vo_grp,payvo1.vo_nur,kpurlo1.kpurl_datrp,kpurlo1.kpurl_cdvo,payvo
1.vo_met' + Chr(13);
//ZVN 06/09/10{
If CheckBox9.Checked then
begin
strSQL := strSQL + 'union all' + Chr(13);
strSQL := strSQL + 'select Month(kpurlo1.kpurl_datup) as M_UP,' + Chr(13);
strSQL := strSQL + 'Year(kpurlo1.kpurl_datup) as Y_UP,' + Chr(13);
strSQL := strSQL + 'payvo1.vo_cdchr,payvo1.vo_cdchrs,sum(kpurlo1.kpurl_Sm)
as kpurl_Sm,pspr.spr_nm,' + Chr(13);
strSQL := strSQL + 'kpux.kpu_rcd,' + Chr(13);
strSQL := strSQL + 'sum(case when BIT_AND( KpuRl_Prz2 , 4 ) = 0 AND
KpuRl_LnkSFEK <> 0 Then 0' + Chr(13);
strSQL := strSQL + ' when BIT_AND( KpuRl_Prz , 8 ) = 0 then kpurl_days
else -kpurl_days end) as kpurl_days,' + Chr(13);
strSQL := strSQL + 'sum(case when BIT_AND( KpuRl_Prz2 , 4 ) = 0 AND
KpuRl_LnkSFEK <> 0 Then 0' + Chr(13);

```

```

        strSQL := strSQL + '      when BIT_AND( KpuRl_Prz , 8 ) = 0 then kpur1_hrs
else -kpurl_hrs end)as kpur1_hrs,' + Chr(13);
        strSQL := strSQL + 'kpurl01.kpur1_datup,payvo1.vo_grp,payvo1.vo_nur,0 AS
kpurlSvm_tn,' + Chr(13);
        strSQL := strSQL + '0 AS kpur1_sf,0 AS
kpurl_ek,kpurlo1.kpur1_datup,kpurlo1.kpur1_cdvo,0 AS svm,payvo1.vo_met' + Chr(13);
        strSQL := strSQL + 'from kpurlo1' + Chr(13);
        strSQL := strSQL + 'inner join payvo1 on payvo1.vo_cd=kpurlo1.kpur1_cdvo' +
Chr(13);
        strSQL := strSQL + 'inner join pspr on payvo1.vo_cd=pspr.spr_cd and
pspr.sprspr_cd=787202' + Chr(13);
        strSQL := strSQL + 'inner join kpux on kpurlo1.kpu_tn=kpux.kpu_tn' + Chr(13);
        strSQL := strSQL + 'where payvo1.vo_grp=130' + Chr(13);
        If CheckBox9.Checked = False then
        begin
            strSQL := strSQL + 'AND BIT_AND( KpuRl_Prz , 65536 ) = 0' + Chr(13);
        end;
        // Вибірково з ІФ
        // Вибірково з ІФ
        If ComboBox5.ItemIndex <> -1 then
        begin
            strSQL := strSQL + 'and
payrld_sf=' + vartostr(ComboBox5.Items.Objects[ComboBox5.ItemIndex]) + ' + Chr(13);
        end;
        // вибірково за ЕКР
        If ComboBox6.ItemIndex <> -1 then
        begin
            strSQL := strSQL + 'and
payrld_ek=' + vartostr(ComboBox6.Items.Objects[ComboBox6.ItemIndex]) + ' + Chr(13);
        end;
        strSQL := strSQL + 'group by
Month(payrldo1.payrld_up),Year(payrldo1.payrld_up),' + Chr(13);
        strSQL := strSQL +
'payvo1.vo_cdchr,payvo1.vo_cdchrs,pspr.spr_nm,kpux.kpu_rcd,payrldo1.payrld_dat,' + Chr(13);
        strSQL := strSQL +
'payrldo1.payrld_up,payvo1.vo_grp,payvo1.vo_nur,payrldo1.rl_cdvo,payvo1.vo_met' + Chr(13);
        end;
        //ZVN 06/09/10 }
        begin
            strSQL := strSQL + 'and
kpurl_ek=' + vartostr(ComboBox6.Items.Objects[ComboBox6.ItemIndex]) + ' + Chr(13);
        end;
        strSQL := strSQL + 'GROUP BY Month(payrldo1.payrld_up),' + Chr(13);
        strSQL := strSQL + 'Year(payrldo1.payrld_up),' + Chr(13);
        strSQL := strSQL +
'payvo1.vo_cdchr,payvo1.vo_cdchrs,pspr.spr_nm,payrldo1.payrld_dat,' + Chr(13);
        strSQL := strSQL + 'kpux.kpu_rcd,' + Chr(13);
        strSQL := strSQL + 'payrldo1.payrld_up,payvo1.vo_grp,payvo1.vo_nur,' + Chr(13);
        strSQL := strSQL + 'payrldo1.rl_Datrp,payrldo1.rl_cdvo,payvo1.vo_met' + Chr(13);
        //ZVN 06/09/10 {
        If CheckBox9.Checked = True then

```

```

begin
  strSQL := strSQL + 'union all' + Chr(13);
  strSQL := strSQL + 'select Month(payrldo1.payrld_up) as M_UP,' + Chr(13);
  strSQL := strSQL + 'Year(payrldo1.payrld_up) as Y_UP,' + Chr(13);
  strSQL := strSQL + 'payvo1.vo_cdchr,payvo1.vo_cdchrs,Sum(payrldo1.rl_sm) as
KpuRl_Sm,pspr.spr_nm,payrldo1.payrld_dat,' + Chr(13);
  strSQL := strSQL + 'kpux.kpu_rcd,' + Chr(13);
  strSQL := strSQL + 'Sum(case when BIT_AND( Rl_Prz2 , 4 ) = 0 AND
Rl_LnkSFEK <> 0 Then 0' + Chr(13);
  strSQL := strSQL + '  when BIT_AND( Rl_Prz , 8 ) = 0 then rl_days else -
rl_days end) as kpurl_days,' + Chr(13);
  strSQL := strSQL + 'Sum(case when BIT_AND( Rl_Prz2 , 4 ) = 0 AND
Rl_LnkSFEK <> 0 Then 0' + Chr(13);
  strSQL := strSQL + '  when BIT_AND( Rl_Prz , 8 ) = 0 then rl_hrs else -rl_hrs
end) as kpurl_hrs,' + Chr(13);
  strSQL := strSQL + 'payrldo1.payrld_up as
kpurl_datup,payvo1.vo_grp,payvo1.vo_nur,' + Chr(13);
  strSQL := strSQL + '0 as kpurl_sf,0 as kpurl_ek,payrldo1.payrld_up as
kpurl_datrp,payrldo1.rl_cdvo as kpurl_cdvo,0 as svm,payvo1.vo_met' + Chr(13);
  strSQL := strSQL + 'from payrldo1' + Chr(13);
  strSQL := strSQL + 'inner join payvo1 on payvo1.vo_cd=payrldo1.rl_cdvo' +
Chr(13);
  strSQL := strSQL + 'inner join pspr on payvo1.vo_cd=pspr.spr_cd and
pspr.sprspr_cd=787202' + Chr(13);
  strSQL := strSQL + 'inner join kpux on payrldo1.payrld_tn=kpux.kpu_tn' +
Chr(13);
  strSQL := strSQL + 'where payvo1.vo_grp=130' + Chr(13);
  If CheckBox9.Checked = False then
    begin
      strSQL := strSQL + 'AND BIT_AND( payrldo1.payrld_prz , 65536 ) = 0' +
Chr(13);
    end;
  // Вибірково з ІФ
  If ComboBox5.ItemIndex<>-1 then
    begin
      strSQL := strSQL + 'and
payrld_sf='+vartostr(ComboBox5.Items.Objects[ComboBox5.ItemIndex])+' + Chr(13);
    end;
  // Вибірково за ЕКР
  If ComboBox6.ItemIndex<>-1 then
    begin
      strSQL := strSQL + 'and
payrld_ek='+vartostr(ComboBox6.Items.Objects[ComboBox6.ItemIndex])+' + Chr(13);
    end;
  strSQL := strSQL + ' group by
Month(payrldo1.payrld_up),Year(payrldo1.payrld_up),' + Chr(13);
  strSQL := strSQL + '
payvo1.vo_cdchr,payvo1.vo_cdchrs,pspr.spr_nm,kpux.kpu_rcd,payrld_dat,' + Chr(13);
  strSQL := strSQL +
'payrldo1.payrld_up,payvo1.vo_grp,payvo1.vo_nur,payrldo1.rl_cdvo,payvo1.vo_met' + Chr(13);
end;

```

```

//ZVN 06/09/10}

strSQL := strSQL + ') tt' + Chr(13);
strSQL := strSQL + 'where tt.kpu_rcd=:rcd' + Chr(13);
strSQL := strSQL + 'and tt.payrld_dat=:DT' + Chr(13);
If CheckBox3.Checked=False then
begin
strSQL := strSQL + 'and tt.vo_nur=0' + Chr(13);
end
// Включати ВО, які не входять до сум
// Нараховано\Утримано
Else If (CheckBox3.Checked) AND (str_vo <> "") then
begin
strSQL := strSQL + 'and (tt.vo_nur=0 or tt.kpurl_cdvo in ('+str_vo+'))' + Chr(13);
end;
strSQL := strSQL + 'group by
tt.M_UP,tt.Y_UP,tt.vo_cdchr,tt.vo_cdchrs,tt.kpurl_cdvo,tt.spr_nm,tt.kpu_rcd,tt.kpurl_days,tt.kpurl_
hrs,tt.kpurl_datup,' + Chr(13);
strSQL := strSQL +
'tt.vo_grp,tt.vo_nur,tt.kpurl_sf,tt.kpurl_ek,tt.kpurl_datrp,tt.payrld_dat,' + Chr(13);
strSQL := strSQL + 'case when tt.vo_grp<=127 then 0 else 1 end,tt.svm,tt.vo_met' +
Chr(13);
end;
end;
If CheckBox4.Checked=False then
begin
strSQL := strSQL + 'order by tt.Y_UP,tt.M_UP,case when tt.vo_grp<=127 then 0 else 1
end,tt.kpurl_datrp,tt.vo_cdchr' + Chr(13);
RL.SQL.TEXT := strSQL;
RL.ParamByName('rcd').DataType:=ftInteger;
RL.ParamByName('rcd').Value:=<Query1."KPU_RCD">;
RL.ParamByName('m').DataType:=ftInteger;
RL.ParamByName('y').DataType:=ftInteger;
RL.ParamByName('m').Value:=<Query1."M">;
RL.ParamByName('y').Value:=<Query1."Y">;
end
// Розрахункові листи на дату
Else
begin
RL.SQL.TEXT := strSQL;
RL.ParamByName('rcd').DataType:=ftInteger;
RL.ParamByName('rcd').Value:=<Query1."KPU_RCD">;
RL.ParamByName('DT').DataType:=ftDate;
RL.ParamByName('DT').Value:=DateEdit1.Date;
end;
end;

procedure GroupFooter2OnAfterPrint(Sender: TfrxComponent);
begin
If (CheckBox7.Checked) then
begin

```

```

BPRO.FILE_WRITE(pDescr,'</table>');
strTxtLetter := strTxtLetter + '</table>' + Chr(13);
If sm_n+sm_s-sm_u >= 0 then
begin
    strTxtLetter := strTxtLetter + '<font size=4 face="Arial">'+ENUA('THE
ENTERPRISE'S DEBT','БОРГ ПІДПРИЄМСТВА НА')+UpperCase(str_m1)+'
'+UpperCase(str_y)+':'+</font>';
    strTxtLetter := strTxtLetter + '<font size=4
face="Arial">'+FormatFloat('#,##0.00',sm_n+sm_s-sm_u)+</font>'+Chr(13);
    BPRO.FILE_WRITENC(pDescr,'<font size=4 face="Arial">'+ENUA('THE
ENTERPRISE'S DEBT','БОРГ ПІДПРИЄМСТВА НА')+UpperCase(str_m1)+'
'+UpperCase(str_y)+':'+</font>');
    BPRO.FILE_WRITE(pDescr,'<font size=4
face="Arial">'+FormatFloat('#,##0.00',sm_n+sm_s-sm_u)+</font>');
end
Else If sm_n+sm_s-sm_u < 0 then
begin
    strTxtLetter := strTxtLetter + '<font size=4 face="Arial">'+ENUA('EMPLOYEE'S
DEBT НА','БОРГ ПРАЦІВНИКА НА')+UpperCase(str_m1)+' '+UpperCase(str_y)+':'+</font>';
    strTxtLetter := strTxtLetter + '<font size=4
face="Arial">'+FormatFloat('#,##0.00',sm_n+sm_s-sm_u)+</font>'+Chr(13);
    BPRO.FILE_WRITENC(pDescr,'<font size=4
face="Arial">'+ENUA('EMPLOYEE'S DEBT НА','БОРГ ПРАЦІВНИКА НА
')+UpperCase(str_m1)+' '+UpperCase(str_y)+':'+</font>');
    BPRO.FILE_WRITE(pDescr,'<font size=4
face="Arial">'+FormatFloat('#,##0.00',sm_n+sm_s-sm_u)+</font>');
end;
strTxtLetter := strTxtLetter + '<hr></hr>'+Chr(13);
BPRO.FILE_WRITE(pDescr,'<hr></hr>');
strTxtLetter := strTxtLetter + '</body>'+Chr(13);
strTxtLetter := strTxtLetter + '</html>'+Chr(13);
BPRO.FILE_WRITE(pDescr,'</body>');
BPRO.FILE_WRITE(pDescr,'</html>');
BPRO.FILE_CLOSE(pDescr);
// Надсилаємо лист із вкладенням поштою
If (CheckBox8.Checked)then
if (<Query1."Kpu_Email"> = " ) then
begin
    if CheckBox10.Checked then
begin
        PrMail.Insert;
        PrMail.FieldByName('Typ').Value:=3;
        PrMail.FieldByName('TN').Value:=<QUERY1."kpu_tn">;
        PrMail.FieldByName('FIO').Value:=<QUERY1."kpu_fio">;
        PrMail.Post;
    end;
end
else
begin
If TRIM(Edit9.Text) <> " then
begin

```

```

        strTheme := Edit9.Text;
    end
Else
    begin
        strTheme := 'Розрахунковий лист за '+TRIM(strМемо21);
    end;
If TRIM(Memo31.Lines.Text) = " then
    begin
        //strTxtLetter := ' ';
    end
Else
    begin
        //strTxtLetter := Memo31.Lines.Text;
    end;
try
    If ( 0 =
BPRO.MAIL_SEND(Edit3.Text,Edit4.Text,Edit2.Text,StrToInt(Edit5.Text),StrToInt(Edit6.Text),E
dit7.Text,Edit8.Text,Copy(<Query1."Kpu_Email">,1,Pos('@',<Query1."Kpu_Email">)-
1),<Query1."Kpu_Email">,strTheme,strTxtLetter,pFilePath) ) then
        begin
            if CheckBox10.Checked then
                begin
                    PrMail.Insert;
                    PrMail.FieldName('Typ').Value:=2;
                    PrMail.FieldName('TN').Value:=<QUERY1."kpu_tn">;
                    PrMail.FieldName('FIO').Value:=<QUERY1."kpu_fio">;
                    PrMail.FieldName('EMAIL').Value:=<Query1."Kpu_Email">;
                    PrMail.Post;
                end;
            end
        else
            begin
                if CheckBox10.Checked then
                    begin
                        PrMail.Insert;
                        PrMail.FieldName('Typ').Value:=1;
                        PrMail.FieldName('TN').Value:=<QUERY1."kpu_tn">;
                        PrMail.FieldName('FIO').Value:=<QUERY1."kpu_fio">;
                        PrMail.FieldName('EMAIL').Value:=<Query1."Kpu_Email">;
                        PrMail.Post;
                    end;
                end;
            finally
                end;
        end;
    end;
end;
end;

procedure Button7OnClick(Sender: TfrxComponent);
var pFrm :Integer;
begin

```

```

pFrm := TFrxDialogPage(FRMPARM).ShowModal;
end;

function CheckMailParams():string;
var sMes: string;
begin
  sMes:="";
  If TRIM(Edit2.Text) = " then sMes:=EnUa('Mail server password not entered!','Не
введено пароль поштового сервера!')+chr(13);
  If TRIM(Edit3.Text) = " then sMes:=sMes+EnUa('Mail server name not entered!','Не
введено імя поштового сервера!')+chr(13);
  If TRIM(Edit4.Text) = " then sMes:=sMes+EnUa('Mail server login not entered!','Не
введено логін поштового сервера!')+chr(13);
  If TRIM(Edit5.Text) = " then sMes:=sMes+EnUa('Port number not entered for
transmission!','Не введено номер порта для передачі!')+chr(13);
  If TRIM(Edit6.Text) = " then sMes:=sMes+EnUa('Secure port number not entered!','Не
введено номер захищеного порта!')+chr(13);
  If TRIM(Edit7.Text) = " then sMes:=sMes+EnUa('Sender name not entered!','Не введено
ім'я відправника!')+chr(13);
  If TRIM(Edit8.Text) = " then sMes:=sMes+EnUa('Sender address not entered!','Не
введено адресу відправника!')+chr(13);
  Result:=sMes;
end;

```