

# НАУКИ ПРО ЗЕМЛЮ. ГЕОІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ МОДЕЛЮВАННЯ СТАНУ ГЕОСИСТЕМ

УДК: 004.9:911.5/.9:528.94 <http://dx.doi.org/10.31548/zemleustriy2022.03.11>

## ГОЛОВНІ КОНЦЕПТУАЛЬНІ ПОЛОЖЕННЯ СТВОРЕННЯ ЕЛЕКТРОННОГО ДЕРЖАВНОГО РЕЄСТРУ КУЛЬТУРНОЇ СПАДЩИНИ УКРАЇНИ. ЧАСТИНА 2: ПРОЦЕСИ

**В.С. ЧАБАНЮК,**

*кандидат фізико-математичних наук,  
старший науковий співробітник,  
Інститут географії Національної академії наук України  
E-mail: chab3@i.ua*

**О.П. ДИШЛИК,**

*виконавчий директор, ТОВ «Геоматичні рішення».  
E-mail: dyshlyk@geomatrica.kiev.ua*

**К.А. ПОЛИВАЧ,**

*кандидат географічних наук, провідний науковий співробітник,  
Інститут географії Національної академії наук України  
E-mail: katelyna.polyvach@gmail.com*

**В.І. ПІОРО,**

*директор,  
ГО «Український центр розвитку музейної справи», м. Київ  
E-mail: pioro@ukr.net*

**І.М. КОЛІМАСОВ,**

*начальник виробництва, ТОВ "Інтелектуальні системи-ГЕО"  
E-mail: kolimasov@ukr.net*

**Ю.В. НЕЧИПОРЕНКО,**

*Управління охорони культурної спадщини та музеїв  
Міністерство культури та інформаційної політики України  
E-mail: julnech@ukr.net*

**Анотація.** Частині 2 описані процеси діяльності зі створення нового сучасного електронного Державного реєстру нерухомої культурної спадщини (КС) України. Вони є частиною методології, що базується на Каркасах Рішень (KaPi) АГІС-КС1, де АГІС-КС1 позначає першу чергу ієрархічно структурованої Атласної Геоінформаційної Системи (АГІС) КС. АГІС загалом складається з чотирьох страт: Операційної ( $\omega$ ), Аплікаційної ( $\alpha$ ), Концептуальної ( $\beta$ ) і Загальної ( $\gamma$ ). Процеси, що розглядаються у статті, відносяться до  $\alpha$ КаPi АГІС-КС1, який визначає діяльність між підсистемами АГІС-КС1 Аплікаційної і Операційної страт. Згадуються також процеси, які відносяться до  $\beta$ КаPi АГІС-КС1, який визначає діяльність між підсистемами АГІС-КС1 Концептуальної і Аплікаційної страт.

KaPi АГІС-КС1 визначається пакетами і відношеннями між ними «петради» Публікації-Продукти-Процеси-Основи-Сервіси АГІС-КС1. Пакети Продукти-Процеси-Основи АГІС-КС1 і відношення між ними називаються головною тріадою KaPi. Ця тріада є основою Головних Концептуальних положень 1-3. Для останніх справедливі такі співвідношення: KaPi.Продукти – положення 1, KaPi.Процеси – положення 2, KaPi.Основи – положення 3.

У Частині 2 рекомендуються процеси розроблення і гарантування якості діяльності зі створення АГІС-КС1 із KaPi АГІС-КС1.Процеси. Ці рекомендації є фактично Головним Концептуальним положенням 2.

**Ключові слова:** Каркас Рішень (KaPi), Атласна Геоінформаційна Система (АГІС), Державний реєстр нерухомої культурної спадщини, процеси розроблення і гарантування якості.

## Вступ

Продовжується розгляд діяльності, яку рекомендується організувати для успішної реалізації проекту створення нового сучасного електронного Державного реєстру нерухомої культурної спадщини (КС) України. Для організації цієї діяльності на початку проекту запропоновано застосувати методологію, яка базується на так званому Каркасі Рішень (KaPi) першої черги цієї Просторової інформаційної системи (PrIS). Перша черга PrIS має бути екземпляром класу Атласних ГеоІнформаційних Систем (АГІС, [1]), що позначається АГІС-КС1.

Замість ускладнення позначень, які вже використовуються у Частині 1 [2] і будуть використовуватись далі, ми надаємо тут додаткові пояснен-

ня до них. Позначення відносяться як до кінцевого результату створення – АГІС-КС1, так і до методології його створення – KaPi АГІС-КС1. У цій статті ми розрізняємо поняття «створення» і «розроблення». Перше найчастіше застосовується разом з поняттям «система». Поняття «розроблення» ми розуміємо як якусь частину поняття «створення». Найчастіше ми застосовуємо його до частини системи, причому, частиною може бути, наприклад, певний стан системи. Тут маємо на увазі, наприклад, стани системи до тестування і після. Саме тому процеси «розроблення» і «гарантування якості» продукції розуміються як процеси «створення». Надіємось, що пояснення допоможуть краще зрозуміти результати усіх частин статті, а також пояснити, чому ми не ускладнюємо позначення.

Пояснення позначень. АГІС-КС1 ∈ АГІС означає, що екземпляр об'єкта АГІС-КС1 належить множині об'єктів АГІС. Ми могли б використати позначення/формулу АГІС-КС1 ∈ {АГІС-КС1}, однак це позначення звужує множину «допустимих» реалізацій АГІС-КС1. Ми могли б позначити клас систем АГІСs і якийсь його переклад на українську мову (АГІ-Си?), однак у інформатиці часто використовується нотація УММ (UML – Universal Modeling Language). Згідно з нею клас об'єктів зазвичай позначається назвою класу у однині АГІС-КС1 шрифтом “bold”, а екземпляр об'єкта – назвою класу у однині, але з підкресленням назви АГІС-КС1.

Трохи складнішим для розуміння є запис «КаРі АГІС-КС1». Результатом застосування цього КаРі можуть бути чотири результати: 1) КаРі АГІС-КС1 – екземпляр АГІС-КС1 може створюватися застосуванням відповідного екземпляра КаРі, 2), КаРі АГІС-КС1 – множина {АГІС-КС1} допустимих екземплярів АГІС-КС1 може створюватися застосуванням відповідного конкретного КаРі, 3) КаРі АГІС-КС1 – допустимий екземпляр АГІС-КС1 може створюватись застосуванням якогось КаРі із множини {КаРі}, 4) КаРі АГІС-КС1 – об'єднання варіантів 2) і 3). Наведені нотації є досить умовними, оскільки поняття КаРі не використовується без поняття предмета X, до якого воно (він) застосовується. Тобто, запис КаРі без предмета X не має окремого значення – використовується тільки як скорочення запису «Каркас Рішень».

Багаторічний досвід розроблення ПрІС, а також теоретичні дослідження [3], дозволяють стверджувати, що для кожної ПрІС, включаючи систему класу АГІС, завжди існує відповідний

ій КаРі. Досить легко пересвідчитись, що у кожному проекті створення ПрІС існує, явно чи неявно, головна тріада КаРі: так звані пакети елементів Продукти-Процеси-Основи ПрІС. Адже у кожному проекті створюється якийсь елемент-продукт: база даних, карта, атлас, ГІС, ПрІС, тощо. А більш-менш складний продукт неможливо створити без процесів його створення.

На жаль, у проектах створення інформаційних продуктів дуже часто основна увага приділяється самому продукту і майже не приділяється уваги процесам його створення. А потім розробники дивуються, чому такий низький процент (20% за різними джерелами) успішної реалізації інформаційних проектів. Нагадаємо визначення дуалізму продукт-процес із Частини 1: «1) без процесу неможливо створити продукт, 2) без продукту процес не має сенсу» [2]. Вмістом Частини 2 є Головне Концептуальне положення 2, яке є процесною частиною основної тріади КаРі АГІС-КС1. Воно формулюється так: «Процеси КаРі АГІС-КС1: Процесом створення АГІС-КС1 має бути портал РІЕ» (Projects Implementation Environment - Середовище виконання проектів).

У Частині 1 [2] статті описані отримані до 2019 року включно Головні Концептуальні положення 0: «Використання відповідного КаРі є обов'язковим для успіху діяльності по створенню нового електронного Державного реєстру нерухомої КС» і 1: «Продукти КаРі X: Першою чергою фінальної системи X має бути АГІС-КС1 як елемент множини допустимих АГІС». АГІС-КС1 має бути елементом множини допустимих АГІС [1] і позначається АГІС-КС1 ∈ АГІС (або {АГІС}), АГІС-КС1 ⊂ АГІС.

До 2019 р. ми не приділяли уваги процесам створення АГІС-КС, оскільки тоді більш важливим для нас було визначитися з представленнями про очікуваний кінцевий продукт. Власне, саме для отримання конкретних представлень про цей продукт було започатковано пілотний проект у Вінниці [4]. АГІС-КС тоді представлялась як продукт за допомогою Концептуального Каркаса (КоКа) АГІС-КС, а процеси створення окремих складових компонентів-продуктів визначалися тоді «другорядними» «міжстратовими»  $\beta$ КаРі і  $\alpha$ КаРі, але не  $\gamma$ КаРі КоКа АГІС-КС.  $\gamma$ КаРі використовуються для систем, загальніших, ніж АГІС-КС. Таких, як наприклад, АГІС-ВТ (ВТ – Великі Території) [5] і просто АГІС [1].

У 2021 році ми виконали проект з умовною назвою «Впровадження в експлуатацію в Міністерстві культури та інформаційної політики України декларативного електронного реєстру КС». Це впровадження виконувалося з допомогою розробленого нами у пілотному проєкті [4] Модуля Декларування (МД) КС. Результати виконаного проєкту змусили нас зробити кілька дуже важливих висновків.

1. У діяльності зі створення АГІС-КС дуже важко розраховувати на застосування якогось наперед визначеного процесу розроблення якогось окремого продукту. Однак описаний у цій частині власний досвід може допомогти у формуванні реального процесу. Адже наявність процесу є обов'язковою; без нього не буде і продукту.

2. Ми ніяк не можемо швидко вплинути на рівень ІТ підготовки Замовника. Інакше, особи, що будуть приймати рішення, будуть мати достатню ІТ підготовку. Можливо, перед прийняттям рішення про створення

поточної черги АГІС потрібно проводити семінар, присвячений очікуваним процесам створення.

У описаній ситуації, з врахуванням результатів Частини 1, вважаємо за потрібне розкрити у Частині 2 наступні питання.

1. Можливі процеси створення не окремих продуктів, а систем, подібних АГІС-КС, АГІС-ВТ і АГІС.

2. Переверені власною практикою процеси розроблення окремих продуктів і їх переваги і недоліки, які мають допомогти вибрати найпотрібніші з врахуванням процесів створення системи.

3. Вступ до питання якщо не методології, то хоча б визначеної стратегії гарантування якості розроблених продуктів/результатів.

У Частині 2 увага приділяється так званім «процесним» положенням, які є дуалістичними до «продуктових» концептуальних положень, сформульованих у 2019 р. і описаних у Частині 1 роботи. Дуалістичність значить, що до кожного з «продуктових» концептуальних положень потрібно мати дуалістичні «процесні» концептуальні положення

Серед «процесних» концептуальних положень на даний момент ми виділяємо дві групи: 1) розроблення; 2) гарантування якості продукції. Обидва процеси відносяться до процесу створення і є частиною дуалізму «продукт ↔ процес» в тому плані, що процес або процеси мають пояснити, як створювати один із продуктів АГІС-КС і АГІС-КСС АГІС-ВТ. Групу процесів гарантування якості продукції ми вибрали як одну з важливих, але мало використовуваних у процесах створення ПрІС. Вони є другою частиною процесів, які показуються у другій, верифікаційній ча-

стині літери “V” у V-моделі створення АГІС-КС1.

### **Процеси створення АГІС-КС1 і її частин**

#### ***Процеси створення АГІС-КС1***

У першому розділі Частини 2 розглядаються процеси створення АГІС-КС1. Як для цього розділу, так і для наступного, важливим є розуміння відмінностей між «системою» і її «частинами». Так, в ПрІС дуже часто не розрізняються навіть такі поняття як програмне забезпечення (ПЗ) ПрІС або програмна технологія для створення/розроблення ПрІС.

Першою проблемою тут є застосування програмних стандартів з розроблення ПЗ у «системному оточенні». Наукова і технічна література насичені описом стандартів і самими стандартами розроблення ПЗ. При створенні системи виникає питання, що використовувати: «системні» стандарти чи стандарти «ПЗ» як частини системи. Стандарти ПЗ давно і успішно створюються і використовуються, а «системні» стандарти стали популярними набагато пізніше. Тому потрібно щось з чимось узгоджувати.

Другою проблемою тут є розуміння відмінностей між ПЗ або програмною технологією та ПрІС, що створюється з їх використанням. У нашій практиці ми часто зустрічалися з фразою «ГІС ArcGIS», яка її автором найчастіше несвідомо розумілась як «ГІС, що створена з використанням програмної технології ArcGIS». На жаль, оригінальну фразу ми чули від осіб, що приймають рішення. Вона фактично означає «купимо ArcGIS і, таким чином, будемо мати ГІС». Проблемою тут є марно витрачені кошти у випадку витрат лише на ПЗ або на-

віть на програмну технологію. Адже ІС є набагато більшим від ПЗ чи програмної технології.

У 2020-2021 роках ми дійшли до висновку, що ні один із відомих процесів створення результуючої ІС/ПрІС не дозволяє створити такі «системні» продукти, як АГІС-КС1СА-ГІС-КССАГІС-ВТСАГІС. Тому нам довелося приділити увагу не тільки процесу (методу), а й більш фундаментальному поняттю - методології створення систем, подібних АГІС. Однак методологія є елементом пакету Основи КаРі АГІС-КС1, який буде описаний за межами Частини 2. При цьому ми не змогли відділити поняття методології від поняття «стратегії» створення/розроблення ІС/ПрІС, яке також потребує прояснення.

Вкажемо на те, що має сенс словосполучення «стратегія методології». Не вдаючись у пояснення, зауважимо, що ми розділяємо дві методології: прескриптивні та дескриптивні. Прескриптивні методології можемо назвати також нормативними або конструктивними. Саме таку методологію ми пропонуємо використовувати для створення АГІС-КС1. У першій стратегії дослідник конструює або будує мета-артефакт інформаційних технологій (ІТ) як загальну концепцію рішення для вирішення класу проблем, а потім застосовує ці рішення у конкретному контексті. Тим самим використовується конструктивний або, інакше, нормативний підхід. У другій стратегії дослідник намагається вирішити конкретну проблему клієнта, будуючи конкретний артефакт ІТ у цьому конкретному контексті. Тим самим використовується дескриптивний підхід. Потім з отриманого досвіду виділяються конструктивні (prescriptive – прескрип-

тивні, нормативні) знання, які формують загальну концепцію рішення для вирішення класу проблем [6].

На думку Ізбаччикова Ю.С. у випадку специфічної людської діяльності, такої як створення/розроблення ІС, термін «методологія» міг би сприйматися або розумітися простіше, однак це не так. Крім вибору стратегії, методологія створення ІС сама по собі є складним поняттям. Для підтвердження цієї думки, розглянемо визначення з: «Методологія створення ІС полягає в організації процесу побудови ІС і в управлінні цим процесом для того, щоб гарантувати виконання вимог як до самої системи, так і до характеристик процесу розроблення».

Методології, технології та інструментальні засоби проектування складають основу проекту будь-якої ІС. Методологія реалізується через конкретні технології та підтримуючі їх стандарти, методики та інструментальні засоби, які забезпечують виконання процесів життєвого циклу ІС».

У стандарті [7] наведено такі визначення:

- життєвий цикл - еволюція системи, продукту, послуги, проекту чи іншої створеної людиною сутності, від концепції до припинення існування,

- модель життєвого циклу - рамки процесів і заходів, пов'язаних із життєвим циклом, які можна розбити на етапи, які є спільним знаменником для спілкування та розуміння (наш переклад з англійської: каркас процесів та діяльності, пов'язаних із життєвим циклом, які можуть бути організовані в стадії, що також виступають загальним орієнтиром для спілкування та розуміння),

- процес - набір взаємопов'яза-

них або взаємодіючих видів діяльності, який перетворює входи на виходи.

У стандарті ДСТУ ISO/IEC/IEEE 15288:2016 (ISO/IEC/IEEE 15288:2015, IDT) усі процеси розбиті на чотири групи (Рис. 1).

Назви деяких процесів з оригіналу ми б переклали інакше, наприклад (див. Рис. 1 і англ. оригінал): Процеси угоди (Agreement processes) – Договірні процеси, Процеси угоди (Acquisition processes) – Процеси придбання, Процес системного аналізування (System analysis process) – Процес аналізу системи, Процес імплементації (Implementation process) – Процес реалізації, Процес передавання (Transition process) – Перехідний процес або Процес випуску продукції.

Методологія розробки ПЗ - сукупність методів, що застосовуються в життєвому циклі ПЗ і мають загальний філософський підхід». На сьогодні існує не так багато методологій, особливо повних, тобто, таких, що враховують усе стадії життєвого циклу ПЗ та ІС. Саме методологія визначає, які мови і системи будуть застосовуватись для створення/розроблення ПЗ/ІС і, багато в чому, рекомендує, який технологічний підхід при цьому буде використаний.

ПрІС/ГІС є спеціальним видом ІС, тому все сказане вище про методології створення/розроблення ІС справджується і для ПрІС/ГІС. На жаль, визначення обох складових відношення ІС ←ПрІС/ГІС неоднозначні\*. Про неоднозначність визначень ГІС багато сказано у роботах [8], [5] і у наведених там посиланнях. Однак для нас більш

---

\*позначає генералізацію у напрямку стрілки. Зворотнє відношення називається спеціалізацією.



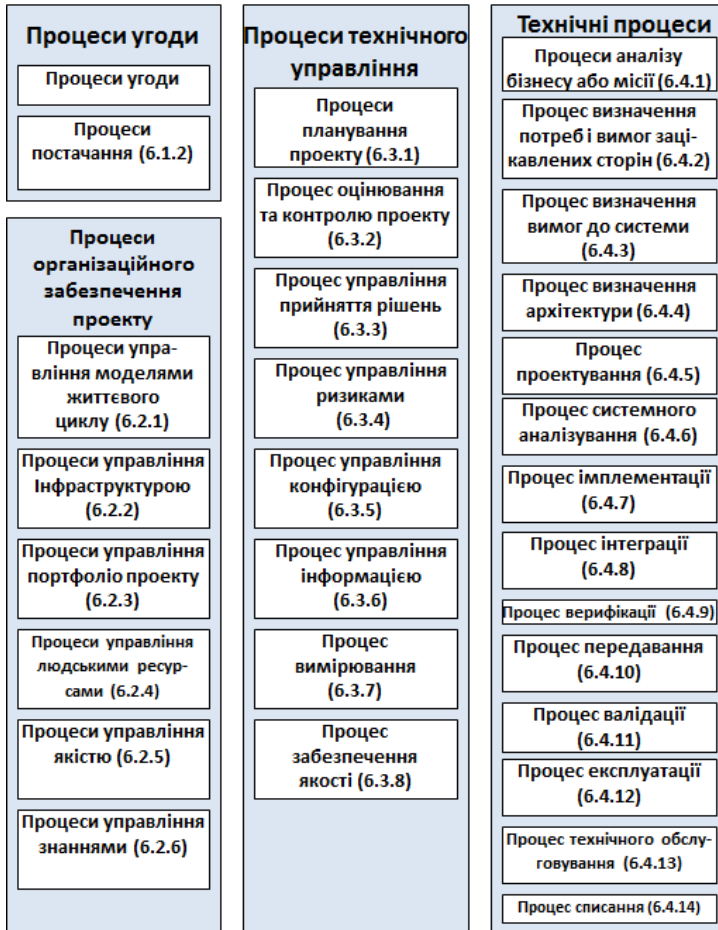


Рис. 1 – Процеси життєвого циклу систем.

важливими є відмінності визначень таких термінів, як ІС, так і ПрІС/ГІС «у вузькому» та у «розширеному» розуміннях. Для цих термінів ми вже давно використовуємо визначення з [9], а саме: 1) Інформаційна система у розширеному розумінні (ІСш) є сукупністю усіх формальних і неформальних представлень даних і дій з ними в організації, включаючи асоційований з першим і другим взаємообмін, як внутрішній, так і з зовнішнім світом; 2) Інформаційна система у

вузькому розумінні (ІСв) є базованими на обчислювальній техніці підсистемами, що призначені забезпечити реєстраційний та підтримуючий сервіси для оперування і управління організацією.

Затонський А.В. основним вибрав визначення ІС, яке, по суті, деталізує визначення ІСш з [11], а «традиційне» визначення ІС є деталізацією визначення ІСв.

«Бізнес-модель - це опис підприємства як складної системи, з заданою

точністю. В рамках бізнес-моделі показані всі об'єкти (сутності), процеси, правила виконання операцій, існуюча стратегія розвитку, а також критерії оцінки ефективності роботи системи. Форма представлення бізнес-моделі і рівень її деталізації визначаються цілями моделювання та прийнятою точкою зору».

«Інформаційна модель - підмножина бізнес-моделі, яка описує всі існуючі (у тому числі і не формалізовані в документальному вигляді) інформаційні потоки на підприємстві, правила обробки та маршрутизації всіх елементів інформаційного поля».

Інформаційна система (ІС) - це вся інфраструктура підприємства, задіяна в процесі управління всіма інформаційно-документальними потоками, включаючи в себе наступні обов'язкові елементи:

- «інформаційна модель, що представляє собою сукупність правил і алгоритмів функціонування ІС: інформаційна модель включає в себе всі форми документів, структуру довідників і даних тощо);

- «регламент розвитку інформаційної моделі і правила внесення в неї змін»;

- «кадрові ресурси (департамент розвитку, залучені консультанти), що відповідають за формування і розвиток інформаційної моделі»;

- «програмний комплекс (ПК), конфігурація якого відповідає вимогам інформаційної моделі (ПК є основним рушієм і, одночасно, механізмом управління ІС); крім цього, завжди існують вимоги до постачальника ПК, що регламентують процедуру технічної і користувацької підтримки впродовж усього життєвого циклу»;

- «кадрові ресурси, що відпо-

відають за конфігурацію ПК і його відповідність затвердженій інформаційній моделі»;

- «регламент внесення змін в конфігурацію ПК і склад його функціональних модулів»;

- «апаратно-технічна база, яка відповідає вимогам з експлуатації ПК (комп'ютери на робочих місцях, периферія, канали телекомунікації), системне програмне забезпечення (ПЗ) і системи управління базами даних (СУБД)»;

- «експлуатаційно-технічні кадрові ресурси, включаючи персонал по обслуговуванню апаратно-технічної бази»;

- «правила використання ПК і призначені для користувача інструкції, регламент навчання і сертифікації користувачів».

За традицією, що склалася, ІС прийнято називати програмно-апаратний комплекс для оброблення (в тому числі і автоматичної) інформації користувача, розробки рішень, формування документів і т.д., що не є коректним, оскільки це лише частина нормальної функціональності ІС. Існує багато визначень поняття системи, але всі вони мають на увазі єдність законів руху (розвитку) складових елементів. Якщо ж ми говоримо про систему, побудовану людиною, то закони руху повинні визначатися конкретними цілями - наприклад, уже перерахованими. А, наприклад, ПЗ за відсутності інформаційної моделі (в контексті даного питання) позбавлене власних законів розвитку і є не більше ніж необхідним інструментом для побудови системи.

ІС забезпечує частину обробки інформації, отже, глобальна мета ІС повинна полягати у забезпеченні її можливо більшого внеску в цілі орга-



нізації через використання інформаційних технологій (ІТ)».

Розроблена нами методологія створення систем класу АГІС базується на патернах і є нормативною. Це значить, що конкретна ПрІС у розширеному або вузькому розумінні створюється з використанням відповідних патернів, а якщо таких поки що не виявлено, то використовуються відповідні нормативні моделі.

Існує три варіанти цієї методології, які ми називаємо: 1) Атласним розширенням AtEx (Atlas Extension, по епістемологічній ієрархії це «знизу-вгору»), 2) ГеоІнформаційним Розширенням ГІР (GIE - GeoInformation Extension, по епістемологічній ієрархії це «згори-вниз»), 3) Комбінованою. Макет методології Атласного розширення AtEx описано у роботі [10].

Майже двадцять років тому ми розробили не тільки метод Каркас ГеоРішень GeoSF (GeoSolutions Framework, GeoSF method, [11]), а й засоби GeoSF (GeoSF means, [12]), див. Помилка! Джерело посилання не знайдено.. У цитованих статтях вказано, що метод за майже 20 років не змінився, а засоби потребують оновлення. Аналізуючи їх, можемо побачити, що у сучасних умовах досить просто оновити порталне програмне рішення TriNet, яке було основою GeoSF means. Тому вважаємо доцільним і можливим реалізацію порталу РІЕ (Projects Implementation Environment - Середовище виконання проєктів) з оновленням порталного рішення та зі спеціалізацією процесів розробки АГІС-КК1. Нагадаємо, що портал (засоби) GeoSF [12] пропонувався майже двадцять років тому для побудови РІЕ виконання групи пов'язаних між собою проєктів.

Спочатку потрібно було інстальовати у мережі intranet/extranet/Інтернет портал GeoSF з початковими значеннями усіх патернів і шаблонів для проєктів, які планувалося виконувати. Створювалися три групи користувачів, відповідальні за різні частини групи проєктів: читачі, автори і координатори, і визначалися конкретні користувачі, що належали до вказаних груп. Після цього РІЕ постійно оновлювалося додаванням нових результатів, які належали до одного з п'яти пакетів GeoSF. Тобто, процес виконання проєктів зводився до ітеративного розроблення РІЕ, а з нею і результуючої ІСш.

У нашому випадку ІСш співпадає з АГІС-КК1. Інтегрована ешелонувана АГІС-КК1, мабуть, буде розроблятися кількома підприємствами. Діяльність цих підприємств можливо розділити на «проєктну» і «щоденну». «Проєктну» діяльність можливо організувати згідно Каркаса проєктних рішень ProSF (Project Solutions Framework). «Непроєктну» або «щоденну» діяльність можливо організувати згідно ComSF (Company SF, GeoSF=ProSFUComSF). Непроєктна діяльність цих підприємств є менш важливою у даному контексті, тому нею можемо знехтувати. Якщо припустити, що АГІС-КК1 є якоюсь частиною НППД – Культурної ППД (КППД), то доцільно згадати 4-й динамічний принцип НППД: «Д4. КоКа КППД як конструктор Просторово уможливленого суспільства (SES) в Україні» [3]. Рис. 2 пояснює застосування цього принципу. Одночасно він пояснює нашу базовану на патернах методологію створення ПрІС загалом і АГІС-КК1 зокрема. Більш повний опис методології знаходиться за межами Частини 2.

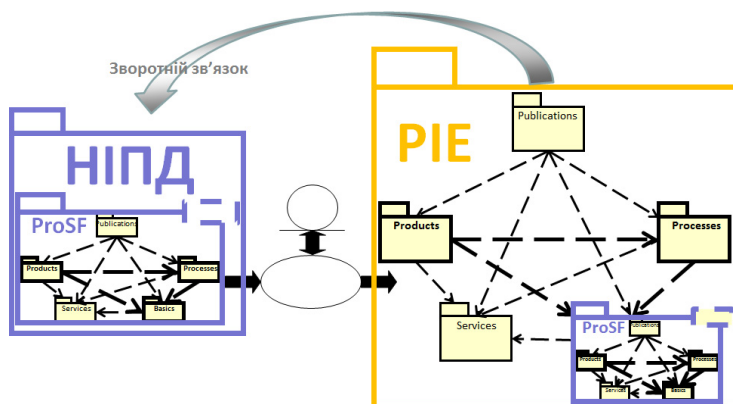


Рис. 2 - Принцип використання ProSF для розроблення порталу PIE.

Нам відома всього одна повна декларативна методологія створення ІС/ПрІС. Вона базується на водоспадній моделі, тому називається «водоспадною». Водоспадна методологія не тільки найвідоміша. Вона дуже важлива ще й тому, що в Україні використання водоспадної моделі є обов'язковим при створенні державних ІС і ПЗ. Раніше існував навіть Єдиний комплекс стандартів і керівних документів на автоматизовані системи, який ми позначали ГОСТ 34.\*\*\*. На даний момент деякі з цих стандартів і документів оновлені і є Міждержавними стандартами. Наприклад, ГОСТ 34.602-2020 – Технічне завдання на створення Автоматизованих системи. Комплекс стандартів на автоматизовані системи.

Маємо на увазі діючу Постанову КМУ від 4 лютого 1998 р. N 121 Київ «Про затвердження переліку **обов'язкових етапів\*\*** робіт під час проектування, засобів інформатизації» {З змінами, внесеними згідно з Постановами КМ N 1758 від 27.12.2001, N 1191 від 11.11.2009, N 675 від 21.07.2010, N 915 від 31.08.2011} [13]. У актуальних назві та тексті

Постанови слова "систем і засобів автоматизованої обробки та передачі даних" [13] замінено словами "засобів інформатизації" [13] згідно з Постановою КМ N 1191. Звідси робимо висновок, що Постанова відноситься і до проекту створення АГІС-КС1. У вказаній Постанові наведено перелік таких обов'язкових етапів (стадій) робіт:

- «Визначення потреби у засобах інформатизації та вивчення питання щодо можливості їх модернізації для забезпечення виконання необхідних функцій» [13].
- «Розроблення техніко-економічного обґрунтування розроблення або модернізації засобів інформатизації» [13].
- «Обґрунтування необхідності використання особистих немайнових та (або) майнових прав інтелектуальної власності на засоби інформатизації» [13].
- «Розроблення технічного завдання на розроблення або модернізацію засобів інформатизації» [13].

\*\*Виділено авторами.

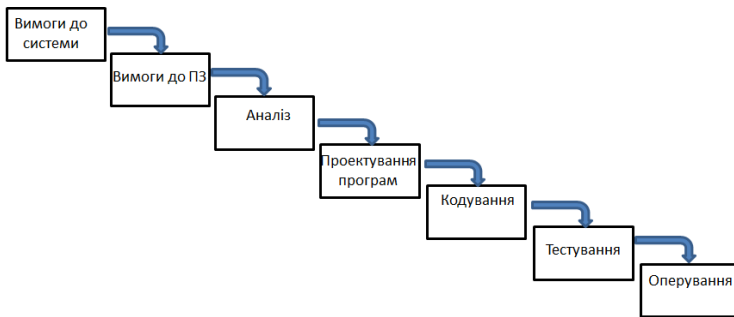


Рис. 3 – Водоспадна модель за [14].

- «Розроблення технічного та робочого або техноробочого проекту розроблення чи модернізації засобів інформатизації» [13].
- «Проведення навчання фахівців для забезпечення функціонування засобів інформатизації» [13].
- «Виконання пусконаладжувальних робіт» [13].
- «Проведення випробувань створених або модернізованих засобів інформатизації» [13].
- «Введення засобів інформатизації в експлуатацію» [13].
- «Виконання робіт з обслуговування засобів інформатизації відповідно до гарантійних зобов'язань» [13].
- «Післягарантійне обслуговування засобів інформатизації» [13].

#### **Процеси розроблення окремих складових АГІС-КС1**

У наступних підрозділах описуються чотири процеси розроблення ПЗ/ІС, які ми використовували на практиці, тому на своїх висновках будемо настоювати. Це, з одного боку, значить, що маємо багато додаткової доказової інформації. З другого боку,

вважаємо за необхідне рекомендувати свої висновки тим, хто буде створювати АГІС-КС1. Можливо, наш досвід допоможе визначитись з процесами створення/розроблення першої черги АГІС-КС.

#### **Водоспадна або Каскадна модель**

Водоспадна модель розроблення ІС/ПЗ називається також Каскадною. В ній процес створення/розроблення виглядає як потік, що послідовно проходить стадії визначення вимог, проектування, реалізації (конструювання і втілення), тестування і налагодження, інтеграції (інсталяції) та підтримки. Застосовується також до розробки ІС. У якості джерела назви і суті моделі часто вказують статтю [14]. Ройс описав основні концепції того, що нині прийнято називати «каскадною» моделлю, і обговорив недоліки цієї моделі. Там же він показав, як ця модель може бути доопрацьованою до ітераційної моделі. У оригіналі було (рис.3).

Перехід від даної стадії до наступної здійснюється тільки після повного і успішного завершення даної стадії. Спочатку повністю завершується ста-

дія «визначення вимог», результатом якої є вимоги до системи загалом і до ПЗ зокрема. Після того, як вимоги повністю визначені, відбувається перехід до аналізу, який варто сприймати як попереднє або концептуальне проектування. Після цього здійснюється логічне проектування, в ході якого створюються документи, що докладно описують програмістам спосіб і план реалізації задокументованих вимог. Після того, як проектування повністю виконане, програмісти виконують реалізацію (кодування) отриманого проекту. Хоча це не показано на Рис. 3, після реалізації відбувається інтеграція окремих компонентів, які у великих проектах часто розробляються різними командами програмістів. Після того, як реалізація та інтеграція завершені, виконуються стадії, які у V-моделі називаються «валідацією» [7, 16]. Зокрема, проводиться тестування та налагодження продукту; на цій стадії усуваються всі недоліки, що виявилися на попередніх стадіях розроблення. Після цього програмний продукт впроваджується та забезпечується його підтримка – внесення нової функціональності та усунення помилок [14, 16].

Тим самим, у каскадній моделі перехід від однієї стадії розроблення до іншої відбувається тільки після повного та успішного завершення попередньої стадії, і що переходів назад або вперед або перекриття стадій не відбувається. «Тим не менш, існують модифіковані каскадні моделі (включаючи модель самого Ройса), що мають невеликі або навіть значні варіації цього процесу» [14].

Можемо говорити не просто про процес, а про методологію створення/розроблення, що базується на каскадній моделі. Цю методологію досить

часто критикують за недостатню гнучкість та оголошення самоціллю формальне управління проектом на шкоду термінам, вартості та якості. Тим не менш, при управлінні великими проектами формалізація часто була дуже великою цінністю, оскільки могла кардинально знизити багато ризиків проекту і зробити його більш прозорим. Тому навіть у PMBOK (Project Management Body of Knowledge) 3-ї версії формально було закріплено лише методологію «каскадної моделі» і не було запропоновано альтернативних варіантів, відомих як ітеративне ведення проектів.

Починаючи з PMBOK 4-ї версії [15] вдалося досягти компромісу між методологами, прихильними до формального та поступального управління проектом, з методологами, які роблять ставку на гнучкі ітеративні методи. Таким чином, починаючи з 2009 року, формально Інститутом управління проектами (PMI) пропонується як стандарт «гібридний варіант методології управління проектами, що поєднує як плюси від водоспадної методології, так і досягнення ітеративних методологів» [15].

### **V-модель розроблення**

Цю модель ми використовували у проекті "Інформаційно-аналітична система забезпечення містобудівної діяльності" (ІАС ЗМД) міста Києва у 2018 році.

Так трапилося, що ми підключилися до проекту «зсередини», одразу до стадії реалізації з обмеженим впливом на такі компоненти проекту як план, команда виконавців, засоби (ArcGIS) тощо. Багато важливих проектних рішень було прийнято до нас і нам довелося їм слідувати. У проекті ми використовували V-модель для

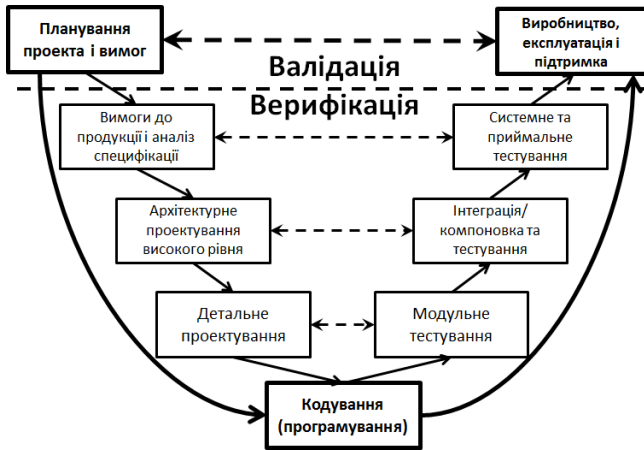


Рис. 4. Спрощений варіант V-моделі розроблення для проєкту ІАС ЗМД

розроблення і гарантування якості. Звісно, V-модель була спрощена, що є наслідком початкового аналізу проєкту (Рис. 4). Зауважимо, що у монографії [16] V-модель описана, а у от у наступній редакції цієї ж монографії опису V-моделі вже не було. Ми не шукали причини видалення з робіт вказаних авторів, однак є такі роботи як “The Death Of The V-Model”, Ed Liversidge, Jun 25, 2015 [17].

На Рис. 4 напівжирним виділені елементи V-моделі, на яких ми концентруватися у проєкті ІАС ЗМД. При цьому така концентрація відносилась як до системи у цілому, так і до кожного окремого модуля, який розроблявся у проєкті. Поняття «валідація» і «верифікація» додатково пояснюють пріоритети проєктувальних робіт проєкту (усіх робіт, крім кодування (програмування)).

Поняття «верифікації» і «валідації» тісно пов'язані з процесами тестування і забезпечення якості. Їх часто плутають, хоча відмінності між ними досить істотні.

«Верифікація (verification) - це процес оцінки системи або її компонентів з метою визначення того, чи задовольняють результати поточної стадії розроблення умовам, сформованим на початку цієї стадії» [15]. Тобто, чи виконуються завдання, цілі та строки розроблення продукту.

Верифікація - підтвердження через надання об'єктивних доказів того, що визначені вимоги виконано.

Валідація (Validation) - підтвердження шляхом надання об'єктивних доказів того, що вимоги щодо конкретного використання за призначенням або заявку виконано.

«Валідація (validation) – це визначення відповідності розроблюваного ПЗ очікуванням і потребам користувача, вимогам до системи» [15] допомагає виділити ключові відмінності між цими поняттями.

За допомогою валідації впевнюються, що створюється «правильний» продукт, який повністю задовольняє замовника. За допомогою верифікації можливо переконатися в тому, що

Табл. 1. Верифікація і валідація

№	Верифікація	Валідація
1	Чи правильно ми створюємо продукт?	Чи правильний продукт ми створюємо?
2	Чи реалізована уся функціональність?	Чи правильно реалізована функціональність?
3	Верифікація виконується раніше і включає перевірку правильності написання документації, коду, тощо	Валідація виконується після верифікації і, як правило, відповідає за оцінку продукту в цілому
4	Виконується розробниками	Виконується тестувальниками
5	Включає статичний аналіз: інспектування коду, порівняння вимог тощо	Включає динамічний аналіз: виконання програми для порівняння її реальної роботи з встановленими вимогами
6	Основається на об'єктивній оцінці відповідності реалізованих функцій	Суб'єктивний процес, що включає особисту оцінку якості роботи ПЗ

продукт створений «правильно»: дотримуючись необхідних методик, інструментів і стандартів.

«На практиці, відмінності верифікації та валідації мають велике значення: замовника цікавить більшою мірою валідація (задоволення власних вимог); виконавця, в свою чергу, хвилює не тільки дотримання всіх норм якості (верифікація) при реалізації продукту, а й відповідність всіх особливостей продукту бажанням замовника» [15]. У зв'язку з верифікацією і валідацією варто взяти до уваги наведене на Рис. 5 представлення V-моделі розроблення.

#### Модель постадійного постачання

Модель постадійного постачання [18] успішно використана нами

у кількох проектах розроблення великих ПрІС. Прикладом є Аналітично-інформаційна система (АІС МТС) другого за величиною оператора мобільного зв'язку в Україні – Vodafone (раніше – МТС).

Постадійне постачання запобігає проблемі водоспадної моделі коли система не розроблена, аж поки усі частини системи не розроблені. Як тільки завершено проектування архітектури, систему можливо розробляти і постачати постадійно. Модель постадійного постачання - це ще одна модель життєвого циклу, у якій ПЗ показується клієнту на стадіях, які послідовно вдосконалюються. На відміну від моделі еволюційного прототипування, коли використовується постадійне постачання, то точно ві-

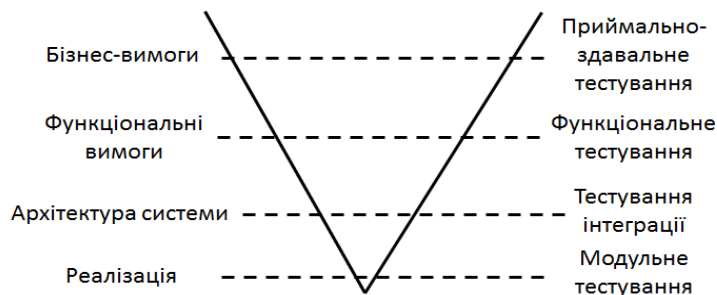


Рис. 5. Рис. 5 – Ще один варіант представлення V-моделі розроблення



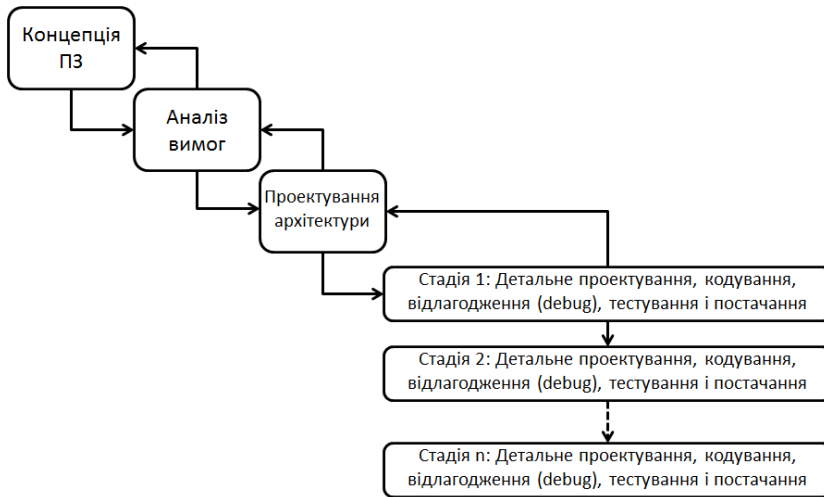


Рис. 6. Модель поетапного постачання

домо, що буде створюватися і коли це буде створено. Особливістю моделі поетапного постачання є те, що ПЗ не постачається наприкінці проекту «одним махом». Воно постачається на послідовних стадіях протягом усього проекту.

На Рис. 6 показано, як працює модель. За допомогою поетапного постачання проходять стадії водоспадної моделі: визначення концепції ПЗ, аналіз вимог і розроблення архітектури всієї програми, яка буде створюватися. Потім здійснюється перехід до детального проектування, кодування, налагодження та тестування на кожній стадії. Основна перевага поетапного постачання полягає в тому, що вона дозволяє передати корисні функції клієнтам раніше, ніж при завершенні 100 відсотків проекту в кінці проекту. Якщо ретельно спланувати стадії, то найважливішу функціональність можливо буде надати якнайшвидше, і клієнти зможуть почати використовувати ПЗ на цій

стадії. Поетапне постачання також забезпечує відчутні ознаки прогресу на ранніх стадіях проекту, ніж менш поступові підходи. Такі ознаки прогресу можуть бути цінним союзником у підтримці тиску графіка на контрольованому рівні.

Основним недоліком поетапного постачання є те, що вона не працюватиме без ретельного планування як на рівні керівництва, так і на технічному рівні. На рівні керівництва переконайтеся, що заплановані стадії важливі для замовника, і що робота розподілена між командою проекту таким чином, щоб вона могла завершити свою роботу вчасно до кінцевого терміну стадії. На технічному рівні потрібно переконаватися, що враховані всі технічні залежності між різними компонентами продукту. Поширеною помилкою є відкладення розроблення компонента до стадії 4 лише для виявлення того, що компонент, запланований для стадії 2, не може працювати без нього.

### Методика Agile

Ми не можемо обійти увагою методику Agile, хоча маємо негативний досвід застосування одного з її методів. Кілька років тому ми мали справу з проектом, де потрібно було виконати черговий «спринт» для існуючої системи. При цьому ми не розробляли архітектуру системи та й відповідної архітектури документації також не було. Ми повелися на видиму простоту додаткової функціональності («спринту») і взялися за проект її реалізації. Функціональне тестування цієї функціональності було успішним. Однак прийнятно-здавальне тестування всієї системи виявило неочікувані нами негаразди. Ми не змогли з ними швидко розібратися і врешті решт провалили проект. У розділі використано інформацію з монографії [19] та інших джерел.

«Agile – система ідей і принципів «гнучкого» управління проектами, на основі яких розроблені популярні методи Scrum, Kanban та інші. Ключовий принцип – розробка через короткі ітерації (цикли), в кінці кожного з яких замовник (користувач) отримує робочий код або продукт» [19].

За появу цієї гнучкої методики розробки відповідальні 17 американських IT-фахівців зі штату Юта. Разом із «Маніфестом гнучкої розробки ПЗ», у якому вперше пролунав термін «Agile», вони прописали 12 принципів Agile-розроблення. Їхня суть зводиться до таких ключових моментів, які визначають характер гнучкої методики розробки:

- «Люди та взаємодія важливіші за процеси та інструменти» [19].
- «Працюючий продукт важливіший за вичерпну документацію» [19].
- «Співпраця із замовником

важливіша за погодження умов контракту» [19].

- «Готовність до змін важливіша за дотримання початкового плану» [19].

«Agile став основою для цілого ряду гнучких методів, серед яких найбільш відомі Scrum, Lean та екстремальне програмування» [19].

«Scrum - метод гнучкої розробки на основі Agile, в основі якого лежить "спринт" - відрізок часу від 1 до 4 тижнів, по закінченню якого повинна бути отримана робоча версія продукту» [19].

«Lean – метод, який виріс на основі системи управління виробництвом Toyota Production System. У його основі – філософія постійного вдосконалення на всіх рівнях організації, де одне з ключових понять – цінність (те, за що готовий платити замовник)» [19].

«Екстремальне програмування (XP) – один з Agile-методів, де важлива роль відводиться періодичному плануванню із залученням замовника. Він дозволяє визначити недоліки попередньої ітерації, пріоритетність завдань, бажану функціональність продукту з урахуванням побажань замовника» [19].

До переваг методики належать:

- «короткі та зрозумілі ітерації – цикли розробки тривають від 2 тижнів до 2 місяців, після закінчення яких замовник отримує робочу версію продукту» [19],
- «високий ступінь залучення виконавців, організаторів та замовників проекту» [19],
- «на чолі кута стоїть робочий продукт як основний показник прогресу – це можна розглядати як плюс, так і мінус, адже в такому разі до команди проекту висуваються високі

вимоги щодо самоорганізації» [19],

- «мінімізація ризиків завдяки гнучкій системі внесення змін» [19],
- «популярність методу серед розробників програм для управління бізнесу» [19].

Недоліки Agile, які органічно «доповнюють» її переваги:

- «стимулювання постійних змін проекту: гнучкість розробки продукту може призвести до того, що він ніколи не дійде до фінальної версії» [19],

- «підвищені вимоги до кваліфікації та досвіду команди: крім безпосередньо розроблення продукту, команда повинна аналізувати можливі способи поліпшення ефективності власної роботи, безперервно обмінюватися інформацією щодо проекту, бути мотивованою та самоорганізованою. Не завжди ресурси проекту дозволяють залучити таких фахівців» [19],

- «філософський характер методики: Agile — це не чітка інструкція до дії, а ціла філософська концепція. Команда не може механічно застосувати механіки «гнучкої» розробки, потрібно прийняти ключові принципи системи» [19],

- «складність підрахунку підсумкової суми роботи: стимуляція змін та вдосконалення кінцевого продукту призводить до плаваючого значення вартості проекту» [19].

### **Гарантування якості продукції**

У стандарті [7] Процеси гарантування якості продукції відносяться до «Технічних процесів» і називаються «Процес верифікації» і «Процес валідації» (Рис. 1). Між ними знаходиться «Процес передавання» (Transition process), який на практиці важко відділити від перших двох. Наприклад,

процеси гарантування якості продукції відносяться до другої частини етапів робіт, які перераховані у Постанові КМУ N 121, хоча побачити це не просто, а саме:

- «Проведення навчання фахівців для забезпечення функціонування засобів інформатизації» [13].

- «Виконання пусконаладжувальних робіт» [13].

- «Проведення випробувань створених або модернізованих засобів інформатизації» [13].

- «Введення засобів інформатизації в експлуатацію» [13].

- «Виконання робіт з обслуговування засобів інформатизації відповідно до гарантійних зобов'язань» [13].

- «Післягарантійне обслуговування засобів інформатизації» [13].

У конкретних проектах ми використовували ті чи інші «доречні» стандарти для визначення різних частин вказаної діяльності. Для організації діяльності зі створення АГІС-КС1 рекомендуємо розпочинати з Операційної концепції, яка готується з використанням стандарту DI-IPSC-81430 «OPERATIONAL CONCEPT DESCRIPTION (OCD)». Вказаний стандарт ми доволі успішно використовували у проекті ІАС ЗМД. Далі у розділі наводяться деякі факти з Робочого проекту / Операційної концепції вказаного проекту.

Звертаємо особливу увагу на використання V-моделі. Маємо сказати тут, що цю модель зарано «списувати» у державних проектах в Україні. Адже V-модель дозволяє упорядкувати водоспадну модель (Постанова КМУ N 121) з процесами гарантування якості. Процеси гарантування якості відносяться до процесів випуску продукції, до яких у реальних проек-

тах «руки не доходять». Використання V-моделі змушує приділяти увагу гарантуванню якості продукту з самого початку проекту.

### Системне гарантування якості

Бажане системне гарантування якості важко забезпечити. Однак у будь-якому випадку рекомендуємо розробляти Паспорт і/або Формуляр існуючої системи/ситуації за ГОСТ Р 59795–2021 (Рис. 7). Для потреб розробки більше підходить Формуляр, однак для покупних компонентів (таких, як ArcGIS) краще підходить Паспорт. Тому вказані документи комбінуються у документі Паспорт/Формуляр існуючої системи/ситуації.

Результатом 1-ї стадії створення ПрІС бажано мати Операційну концепцію. У 1-му змістовному розділі Операційної концепції описують існуючу систему або ситуацію. Основними у цьому описі мають бути Бізнес-вимоги, Функціональні вимоги і Архітектура системи.

З використанням опису існуючої системи/ситуації потрібно виконати Проектування користувачьких приймальних тестів. Зауважимо, що при нормальній розробці по V-моделі стрілка від Функціональних вимог повернута навпаки, стрілки від Архітектури системи не існує. Тому використано червоний колір.

Наступним кроком має бути Приймальне тестування (постфактум, будь яким можливим способом); при виконанні якого потрібно також «заглянути» в Реалізацію (програмування) і використати результат у Приймальному тестуванні (стрілка від Реалізації).

З залученням Бізнес-вимог, Функціональних вимог і Архітектури системи готуються підрозділи 2.9.2-2.9.6 Формуляра (і/або Паспорта) системи. З використанням результатів Приймального тестування готуються підрозділи 2.9.7-2.9.8 Формуляра (і/або Паспорта) системи. Вміст цих підрозділів фактично співпадає (відпо-

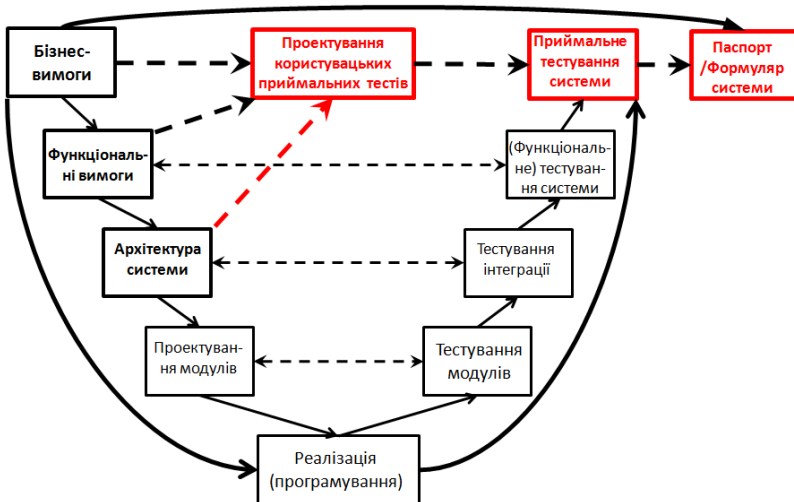


Рис. 7. Системне гарантування якості (на прикладі проекту ІАС ЗМД)

відає) вмісту 2-го змістовного розділу Операційної концепції, де описується обґрунтування і суть змін існуючої системи/ситуації. Отриманий в результаті Паспорт/Формуляр системи можемо вважати навіть модернізацією архітектури, оскільки виготовлення цього документу напевне призведе як до технічних, так і до організаційних змін системи.

Рис. 7 «читається» наступним чином:

Червоний колір виділяє елементи, які потрібно виконати у проекті стосовно існуючої системи. Відновлювані заднім числом Бізнес-вимоги, Функціональні вимоги і Архітектура системи червоним кольором не виділяються, оскільки вони входять в Операційну концепцію і/або Паспорт/Формуляр системи.

### Модульне гарантування якості

Модульне гарантування якості на прикладі проекту ІАС ЗМД пояснює Рис. 8. Воно виконується для функціональних комплексів задач (ФКЗ або модулів), які розроблялися у проекті: присвоєння поштових адрес, присвоєння будівельних адрес, видачі довідок про перейменування вулиць, видача витягу з містобудівного кадастру, видача будівельного паспорта забудови земельної ділянки.

Поняття Паспорт модуля схоже на поняття Паспорт із ГОСТ Р 59795–2021, однак з нього виділено у «окреме виробництво» Приймальне тестування модуля і його наслідки, які фіксуються у окремих документах.

Приймальне тестування модуля включає в себе як валідацію, так і верифікацію модуля, однак перевага надається валідації.

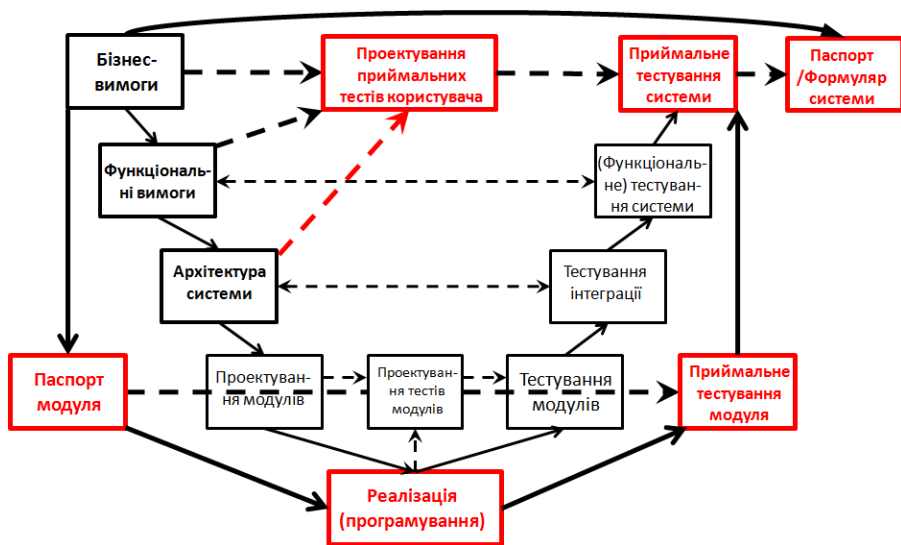


Рис. 8. Рис. 8 - Модульне гарантування якості у проекті ІАС ЗМД, що доповнює Системне гарантування якості

Останнім, третім елементом так званого Паспортного комплексу модуля є Реалізація (програмування), що включає в себе стандарт оформлення коду організації-розробника, а також сам код, оформлений згідно цього стандарту.

### **Висновки.**

Висновки до Частини 2 є рекомендаціями, що обґрунтовані у викладеному матеріалі. Рекомендації пов'язані одна з одною.

1. Для створення АГІС-КС1 потрібно використовувати базовану на патернах методологію «Каркас рішень АГІС-КС1». Елементи методології описані у Частинах 1 і 2 даної статті.

2. Процеси створення АГІС-КС1 вибирати з використанням стандарту ДСТУ ДСТУ ISO/IEC/IEEE 15288:2016.

3. При виборі процесів створення АГІС-КС1 в цілому або її важливих системних компонентів надавати перевагу V-моделі.

4. При виборі процесів розроблення неіснуючого ПЗ використовувати модель постадійного постачання. При виборі існуючого ПЗ надавати перевагу ПЗ з відкритим кодом (open source).

5. Створення концепції АГІС-КС1 і її архітектури є обов'язковими. Для концепції рекомендується зміст стандарту DI-IPSC-81430.

6. Засобами координації має бути набір конкретних шаблонів, патернів і каркасів, організованих у портал. Портал PIE (Projects Implementation Environment – Середовище виконання проєктів) має бути аналогічним порталу GeoSF [15]. Портал поступово (з прогресом виконання проєкту) наповнюється результатами проєкту, які отримуються згідно наперед визначених шаблонів, патернів і каркасів.

7. Потрібно якщо не метод/методологія, то хоча б визначена стратегія гарантування якості отриманих продуктів/результатів.

### **Список літератури**

1. Руденко, та ін. (2018) Культурна спадщина в Атласній геоінформаційній системі сталого розвитку України: Л.Г. Руденко, К.А. Поливач, В.С. Чабанюк та ін./ за ред. Л.Г. Руденка.- Київ: Інститут географії НАН України, 2018.- 172 с.
2. Чабанюк В., та ін. (2022). Головні концептуальні положення розроблення електронного державного реєстру нерухомої культурної спадщини України. Частина 1.- Землеустрій, кадастр і моніторинг земель, № 2, 133-154.
3. Чабанюк В. (2018) Реляційна картографія: Теорія та практика.- Київ: Інститут географії НАН України, 2018.- 525 с.
4. Міністерство культури України. Пілотний проєкт впровадження електронного обліку об'єктів культурної спадщини (URL: <https://www.kmu.gov.ua/ua/npas/pro-realizaciyu-pilotnogo-proektu-vprovadzheniya-elektronnogo-obliku-obyektiv-kulturnoyi-spadshchini>, дата звернення 2022-лип-25).
5. Chabaniuk V., Polyvach K. (2020) Critical properties of modern geographic information systems for territory management.- Cybernetics and Computer Engineering, 2020, № 3(201), pp. 5-32.
6. Iivari J. Distinguishing and contrasting two strategies for design science rese-arch.- European Journal of Information Systems (2015), Vol. 24, Iss. 1, pp. 107-115.
7. ДСТУ ISO/IEC/IEEE 15288:2016 (ISO/IEC/IEEE 15288:2015, IDT). Інженерія систем і програмного забезпечення. Процеси життєвого циклу систем. System Engineering and Software. System Life Cycle Processes.- Національний стандарт України.- 83 (89) с.



8. Chabaniuk V., Kolimasov I. (2020) Analysis of the Practical Use of Geoinformation Systems for Territorial Management and Determination of their Critical Properties.- *Cybernetics and Computer Engineering*, 2020, № 2(200), pp. 5-26.
9. Falkenberg E.D., Lindgreen P., Eds. (1989) *Information System Concepts: An In-depth Analysis*.- Amsterdam et al., North-Holland, 1989.- 357 p.
10. Chabaniuk V., Rudenko L. (2019) Relational geospatial technologies: background theory, practical example and needs in education, pp. 63-83 // in *Geospatial Technologies in Geography Education*. Edited by: de Miguel González Rafael, Donert Karl, Koutsopoulos Kostis.- Springer.- 219 p.
11. Chabaniuk V., Dyshlyk O. (2018) *GeoSolutions Framework Reinvented: Method*, pp. 115-138 // in *Analysis, Modeling and Control*. Vol. 3, Collection of Scientific Papers of the Department of Applied Nonlinear Analysis. Edited by prof. Makarenko A.S.- Institute for Applied System Analysis at the Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, 2018.- 250 p.
12. Chabaniuk V. (2018) *GeoSolutions Framework Reinvented: Means (Portal Realization)*, pp. 90-114 // in *Analysis, Modeling and Control*. Vol. 3, Collection of Scientific Papers of the Department of Applied Nonlinear Analysis. Edited by prof. Makarenko A.S.- Institute for Applied System Analysis at the Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, 2018.- 250 p.
13. Постанова КМУ N 121 від 4.2.1998 {із змінами N 1758 від 27.12.2001, N1191 від 11.11.2009, N 675 від 21.07.2010, N 915 від 31.08.2011}, <https://zakon.rada.gov.ua/laws/show/121-98-%D0%BF#Text> , доступ 2022-вер-22.
14. Royce Winston W. (1987) *Managing the Development of Large Software Systems*.- ICSE '87: Proceedings of the 9th international conference on Software Engineering, March 1987, pp. 328–338.
15. PMBOK Guide (2008) *A guide to the Project Management Body of Knowledge (PMBOK® Guide, 4th Ed.)*. An American National Standard ANSI/PMI 99-001-2008.- Project Management Institute, 2008.- 459 (496) p.
16. Pressman Roger S. (2010) *Software Engineering: A Practitioner's Approach*.- (McGraw-Hill Series In Computer Science).- McGraw-Hill, 7th Ed, 2010.- 895 p.
17. "The Death Of The V-Model", Ed Liversidge, Jun 25, 2015, доступ 2022-вер-23, <https://harmonicss.co.uk/project/the-death-of-the-v-model/>
18. McConnell Steve (1996) *Rapid Development: Taming Wild Software Schedules*.- Microsoft Press, 1996.- 647 (672) p.
19. Pressman Roger S., Maxim Bruse R. (2015) *Software Engineering: A Practitioner's Approach*.- (McGraw-Hill Series In Computer Science).- McGraw-Hill, 8th Ed, 2015.- 941 (971) p.

---

### References

1. Rudenko, at al., (2018) *Kulturna spadshchyna v Atlasnii heoinformatsiinii systemi staloho rozvytku Ukrainy*: L.H. Rudenko, K.A. Polyvach, V.S. Chabaniuk ta in. / Za red. L.H. Rudenka.- Kyiv: Instytut heohrafii NAN Ukrainy, 2018.- 172 s. (Ukrainian)
2. Chabaniuk Viktor, Dyshlyk Oleksandr, Polyvach Kateryna, Pioro Vlad, Kolimasov Ivan, Nechyporenko Julia (2022). *Main Conceptual Provisions of the Creation of an Electronic State of Immovable Cultural Heritage of Ukraine*. Part 1.- Land management, cadastre and land monitoring, No. 2, 30 p. (Ukrainian, English)
3. Chabaniuk V. (2018) *Relational cartography: Theory and practice*.- Kyiv: Institute of Geography of the NAS of Ukraine, 2018.- 525 p. (in Ukrainian)
4. Ministerstvo kultury Ukrainy. *Pilotnyi proekt vprovadzhennia elektronnoho ob-*

- liku ob'ektiv kulturnoi spadshchyny (URL: <https://www.kmu.gov.ua/ua/npas/pro-realizaciyu-pilotnogo-proektu-vprovadzhennya-elektronnogo-obliku-obyektiv-kulturnoyi-spadshchini>, accessed 2022-jul-25).
5. Chabaniuk V., Polyvach K. (2020) Critical properties of modern geographic information systems for territory management.- *Cybernetics and Computer Engineering*, 2020, № 3(201), pp. 5-32.
  6. Iivari J. Distinguishing and contrasting two strategies for design science research.- *European Journal of Information Systems* (2015), Vol. 24, Iss. 1, pp. 107–115.
  7. DSTU ISO/IEC/IEEE 15288:2016 (ISO/IEC/IEEE 15288:2015, IDT). *Inzheneriia system i prohramnoho zabezpechennia. Protsey zhyttievoho tsykladu system. System Engineering and Software. System Life Cycle Processes.- Natsionalnyi standart Ukrainy.- 83 (89) s.*
  8. Chabaniuk V., Kolimasov I. (2020) Analysis of the Practical Use of Geoinformation Systems for Territorial Management and Determination of their Critical Properties.- *Cybernetics and Computer Engineering*, 2020, № 2(200), pp. 5-26.
  9. Falkenberg E.D., Lindgreen P., Eds. (1989) *Information System Concepts: An In-depth Analysis.- Amsterdam et al., North-Holland, 1989.- 357 p.*
  10. Chabaniuk V., Rudenko L. (2019) Relational geospatial technologies: background theory, practical example and needs in education, pp. 63-83 // in *Geospatial Technologies in Geography Education*. Edited by: de Miguel González Rafael, Donert Karl, Koutsopoulos Kostis.- Springer.- 219 p.
  11. Chabaniuk V., Dyshlyk O. (2018) *GeoSolutions Framework Reinvented: Method*, pp. 115-138 // in *Analysis, Modeling and Control*. Vol. 3, Collection of Scientific Papers of the Department of Applied Nonlinear Analysis. Edited by prof. Makarenko A.S.- Institute for Applied System Analysis at the Igor Sikorsky Kyiv Politechnic Institute, Kyiv, 2018.- 250 p.
  12. Chabaniuk V. (2018) *GeoSolutions Framework Reinvented: Means (Portal Realization)*, pp. 90-114 // in *Analysis, Modeling and Control*. Vol. 3, Collection of Scientific Papers of the Department of Applied Nonlinear Analysis. Edited by prof. Makarenko A.S.- Institute for Applied System Analysis at the Igor Sikorsky Kyiv Politechnic Institute, Kyiv, 2018.- 250 p.
  13. Постанова КМУ N 121 від 4.2.1998 {із змінами N 1758 від 27.12.2001, N1191 від 11.11.2009, N 675 від 21.07.2010, N 915 від 31.08.2011} <https://zakon.rada.gov.ua/laws/show/121-98-%D0%BF#Text>, доступ 2022-вер-22.
  14. Royce Winston W. (1987) *Managing the Development of Large Software Systems.- ICSE '87: Proceedings of the 9th international conference on Software Engineering*, March 1987, pp. 328–338.
  15. *PMBOK Guide (2008) A guide to the Project Management Body of Knowledge (PMBOK® Guide, 4th Ed.)*. An American National Standard ANSI/PMI 99-001-2008.- Project Management Institute, 2008.- 459 (496) p.
  16. Pressman Roger S. (2010) *Software Engineering: A Practitioner's Approach.- (McGraw-Hill Series In Computer Science).- McGraw-Hill, 7th Ed., 2010.- 895 p.*
  17. "The Death Of The V-Model", Ed Liversidge, Jun 25, 2015, доступ 2022-вер-23, <https://harmonicss.co.uk/project/the-death-of-the-v-model/>
  18. McConnell Steve (1996) *Rapid Development: Taming Wild Software Schedules.- Microsoft Press, 1996.- 647 (672) p.*
  19. Pressman Roger S., Maxim Bruse R. (2015) *Software Engineering: A Practitioner's Approach.- (McGraw-Hill Series In Computer Science).- McGraw-Hill, 8th Ed, 2015.- 941 (971) p.*

**Chabaniuk V., Dyshlyk O., Polyvach K., Pioro V., Kolimasov I., Nechyporenko J.**  
**MAIN CONCEPTUAL PROVISIONS OF THE CREATION OF AN ELECTRONIC STATE REGISTER OF IMMOVABLE CULTURAL HERITAGE OF UKRAINE. PART 2**

LAND MANAGEMENT, CADASTRE AND LAND MONITORING 3'22: 114-136.

<http://dx.doi.org/10.31548/zemleustriy2022.03.11>

**Abstract.** Part 2 describes the processes of activities for the creation of a new modern electronic State Register of Immovable Cultural Heritage (CH) of Ukraine. They are part of the methodology based on AGIS-CH1 Solutions Framework (SoFr), where AGIS-CH1 represents the first queue of the hierarchically structured Atlas GeoInformation System (AGIS). AGIS-CH generally consists of four strata: Operational ( $\omega$ ), Application ( $\alpha$ ), Conceptual ( $\beta$ ) and General ( $\gamma$ ). The processes in the article refer to AGIS1  $\alpha$ SoFr, which determines the activities "between" AGIS1 subsystems of the Application and Operational strata. The processes related to  $\beta$ SoFr AGIS-CH1, which determines the activities "between" AGIS-CH1 subsystems of the Conceptual and Application strata, are also mentioned.

AGIS-CH1 SoFr is defined by packages and relations between them of Publications-Products-Processes-Basics-Services "petrad". Packages Products-Processes-Basics and the relations between them are called the main triad of SoFr. This triad is the basis of the main conceptual provisions 1-3. They are formulated as follows: AGIS-CH1 SoFr.Products – provision 1, AGIS-CH1 SoFr.Processes – provision 2, AGIS-CH1 SoFr.Basics – provision 3.

Part 2 describes AGIS-CH1 SoFr.Processes – development and quality assurance processes of AGIS-CH1, which corresponds to the Main Conceptual Provision 2.

**Key words:** Solutions Framework (SoFr), Atlas GeoInformation System (AGIS), State Register of Immovable Cultural Heritage, development and quality assurance processes.

---