

**Список використаних джерел:**

1. Ben Cole «IT project management. What is IT project management?». April, 2015 – <https://www.techtarget.com/searchcio/definition/IT-project-management>
2. Agile Manifesto. Manifesto for Agile Software Development, 2001. – <http://agilemanifesto.org>
3. Esther Cohen «The Definitive Guide to Project Management Methodologies» April 25, 2022 – <https://www.workamajig.com/blog/project-management-methodologies>
4. Костюк Н. С. Аналіз методологій управління проектами в ІТ сфері / Костюк Н. С. // Нові компетенції для Індустрії 5.0 та управління даними для закладів вищої освіти : збірник матеріалів круглого столу / під заг. ред. Храпкіної В. В., Пічик К. В.; Національний університет "Києво-Могиллянська академія" [та ін.]. - Київ : НаУКМА, 2023. - С. 65-76.

**УДК 338.22**

**Пелех О.Б.**

доктор економічних наук, доцент,  
професор кафедри менеджменту  
Рівненський державний гуманітарний університет

## **АНАЛІЗ МОДЕЛЕЙ ЖИТТЄВИХ ЦИКЛІВ ПРОЄКТІВ ІНФОРМАТИЗАЦІЇ**

Узагальнена модель життєвого циклу проєкту інформатизації представлена трьома фазами: 1) розроблення стратегії; 2) розроблення та впровадження системи; 3) супровід проєкту. Співвідношення між фазами, їхня повторюваність можуть змінюватися. Тому розроблено декілька моделей життєвого циклу, які відрізняються структурою та конкретним змістом етапів розроблення та впровадження ІС. Відомі на сьогодні моделі утворюють широкий модельний ряд, на протилежних кінцях якого дві моделі – каскадна та спіральна.

Каскадна модель [1] характеризується структурою впорядкованих етапів, найважливішими з яких є етапи створення та впровадження. Ця впорядкованість передбачає, що всі заплановані заходи мають бути виконані настільки ретельно, щоб не виникало потреби в подальшому перегляді рішень. Модель включає лише цикл на етапі технічного обслуговування.

Переваги каскадної моделі: – детермінованість; – суворе регламентація, що спрощує управління проектами і контроль за ними. Недоліки каскадної моделі: – між затвердженням технічного завдання та впровадженням готового продукту проходить тривалий час, тому є ризик, що вимоги користувачів зміняться і не будуть задоволені; – відомі випадки, коли фактичні потреби залишаються незмінними, але були неправильно витлумачені або неадекватно інтерпретовані користувачем під час розроблення технічного завдання.

Спіральна модель [2] передбачає багаторазове проходження одних і тих же етапів проекту доти, поки він не буде задовольняти клієнта. Ця модель відображає ітераційний характер, притаманний процесу розроблення таких складних проектів, як програмне забезпечення ІС. На кожній ітерації створюється робочий прототип, який критично оцінюється. На фінальній ітерації прототип приймається як остаточна версія системи.

Переваги – брак недоліків каскадної моделі, оскільки є можливість пристосування до вимог, що змінюються. Недоліки – складність планування та організації робіт, значні ресурсні витрати під час розроблення великих проектів. Використовується для невеликих проектів, коли є багато невизначеності щодо вимог користувача.

Якщо проект великий, у ньому зазвичай виділяють обмежену підсистему, яку варто розвивати за спіральною моделлю. Як правило, ця модель використовується у випадках, коли замовник, розробник і користувач є однією і тією ж особою або продукт призначений для масової аудиторії.

Проміжне місце між розглянутими моделями належить моделям [2], [3]:

1. Метод швидкого прототипу полягає в розроблення в стислі строки чинного макету частини ІС, найбільш критичної до зміни вимог користувача, проведенні далі дослідної експлуатації пакета з подальшим переходом до розроблення основного зразка. Як правило, користувальницький інтерфейс для майбутніх змін прототипується. Це дає змогу користувачеві бути залученим до проекту на ранній стадії та уникнути дорогого доопрацювання фінальних змін. Основна мета методики – полегшити ідентифікацію вимог користувачів.

2. Метод послідовного нарощування функцій передбачає розроблення та впровадження системи поетапно, зі збільшенням

---

кількості функцій на кожному етапі. Ця модель дає змогу скоротити час впровадження. Тому користувач швидше починає відчувати переваги автоматизації. Очевидною перевагою є скорочення терміну окупності. Слабка сторона – складність планування та управління в поєднанні з необхідністю дотримання відкритої архітектури. Цей метод доцільно використовувати для управлінських ІС для вирішення простих інформаційних завдань, реалізація яких може одразу дати помітний ефект.

3. Еволюційна модель передбачає вдосконалення ІС до рівня якості, який задовольняє кінцевого користувача безпосередньо в процесі дослідної експлуатації. Впровадження ІС починається з функцій, про які розробники мають чітке уявлення. Знання інших функцій системи визначається після її часткового впровадження в експлуатацію. Під час розроблення складної ІС еволюційний підхід дає змогу від самого початку зосередити увагу на досягненні високих експлуатаційних характеристик (надійність, мобільність, модифікованість). Доцільно використовувати підхід при розробленні ІС, коли роботи з розроблення ПЗ не лежать на критичному шляху графіка робіт.

4. Модель, що ґрунтується на повторному використанні компонентів – це основа «складального» програмування, яка дає змогу істотно знизити вартість, тривалість розроблення ІС, підвищити її надійність під час експлуатації. Найбільший ефект спостерігається у випадках, коли значна частина завдань може бути сформульована у вигляді підзадач, які реалізуються стандартними підпрограмами. Тоді розроблення наступної задачі зводиться до написання порівняно простої програми, яка викликає підпрограми в певному порядку й організовує обмін даними між ними. Ця модель є ідеалізацією і практично не використовується в чистому вигляді.

5. Модель, яка ґрунтується на автоматичному синтезі програм, передбачає трансляцію спеціально розроблених програм на мові високого рівня в машинні програми. У сучасному розумінні це поняття базується на знаннях як предметної області, так і процесу розроблення програмних засобів. Такий підхід вимагає високих початкових витрат на побудову моделі знань і особливо на створення інструментарію для їхньої підтримки, що збільшує вартість розроблення. Водночас автоматизований синтез програм дає змогу

різко скоротити всі види витрат, пов'язаних із кожним наступним зразком ІС, і реалізувати високоякісний програмний продукт.

При виборі моделі життєвого циклу інформаційної системи необхідно порівняти сильні та слабкі сторони моделей. Вибір залежить і від замовника ІС. Якщо це ринок або замовник не є державною організацією, то вибір диктується лише логікою здорового глузду; якщо проєкт створюється для держави, то необхідно слідувати ДСТУ, тобто використовувати каскадну модель.

**Список використаних джерел:**

1. *Проектування інформаційних систем: Загальні питання теорії проектування ІС: навч. посіб. / КПП ім. Ігоря Сікорського; уклад.: О. С. Коваленко, Л. М. Добровська. К: КПП ім. Ігоря Сікорського, 2020. 78-84 с.*

2. *Авраменко В.С., Авраменко А.С. Проектування інформаційних систем: навчальний посібник. Черкаси: ЧНУ ім. Б. Хмельницького, 2017. С.29-37.*

3. *Моделі життєвого циклу, принципи і методології розробки програмного забезпечення (ПЗ). <http://surl.li/autcw>*

**УДК 658:005.5, 65.012, 005.4**

**Радіонова Є.В.**

*здобувач вищої освіти*

*Національний університет «Києво-Могилянська академія»*

## **ЗНАЧЕННЯ ЕФЕКТИВНОГО УПРАВЛІННЯ ІТ-ПРОЕКТАМИ ДЛЯ ПІДПРИЄМСТВА**

У сучасному бізнес-середовищі, що швидко розвивається, роль інформаційних технологій (ІТ) стає все більш значущою у досягненні організаційного успіху. Згідно з даними Асоціації «ІТ Україна», у 2022 році ІТ-сектор сприяв надходженню валюти в українську економіку на суму \$7,34 мільярдів. В порівнянні з передвоєнним 2021 роком, обсяг експорту зріс на \$400 мільйонів, що становить приріст у 5,8% [1]. Ефективне управління ІТ-проєктами має першорядне значення для підприємств, які прагнуть використовувати можливості ІТ для збереження конкурентних переваг. У даній роботі описано важливість управління ІТ-проєктами, ключові виклики та стратегії для досягнення успіху на основі таких джерел, як Інститут управління