**Тема 3. Стандартизація розробки програмного забезпечення**

**Міжнародні стандарти ISO. Стандарти організації IEEE – Institute of Electrical and Electronics Engineers.**

Розвиток будь-якої галузі економіки обов'язково супроводжується формалізацією підходів, що використовуються, і появою стандартів різного рівня. На ранніх етапах окремі підприємства формалізують внутрішні процеси, що забезпечує повторюваність результатів процесу або створення певного продукту.

Для полегшення взаємодії підприємств і зручності споживачів розробляються *галузеві стандарти*. Розвиток кожного виду господарської діяльності призводить до потреби в державних засобах забезпечення якості продукції або процесу, тому розробляються і затверджуються *державні стандарти.*

Для поліпшення умов співпраці, розробки загальнозрозумілих правил конкуренції на міжнародному ринку створюються *об'єднання галузевих органів стандартизації,* результатом діяльності яких є *регіональні стандарти* (діють в обмеженому переліку держав, що приєдналися) або *міжнародні стандарти*.

Одним з перших стандартів, що мав істотний вплив на розвиток теорії проектування та розробки ІС, був *стандарт BSP (Business System Planning).* Даний стандарт був розроблений компанією IBM в середині 70- х рр.

Процес BSP передбачав виділення в ході розробки ІС таких кроків:

− отримання підтримки керівництва;

− визначення процесів підприємства;

− визначення класів даних;

− проведення інтерв'ю;

− обробка та організація результатів інтерв'ю.

Найважливіші кроки процесу BSP спостерігаються у більшості формальних методик. На сьогодні діють такі стандарти, що регламентують процес розробки ПЗ:

*ГОСТ* ***34.601-90*** - державний стандарт, поширюється на автоматизовані системи і встановлює стадії і етапи їх створення. У стандарті міститься опис змісту робіт на кожному етапі.

***ISO / IEC 12207*** - Systems and software engineering - Software Life Cycle Processes - міжнародний стандарт на процеси розробки та організації життєвого циклу ПЗ. Поширюється на всі види замовленого ПЗ. Стандарт не містить опису фаз, стадій і етапів.

***SWEBOK (Guide to the Software Engineering Body of Knowledge)*** - Посібник до зведення знань з програмної інженерії - галузевий стандарт Інституту інженерів по радіоелектроніці та електротехніці (IEEE), що містить основні види діяльності з програмної інженерії.

**1.** Міжнародні стандарти **ISO**

Базовим стандартом розробки ПЗ *є ISO 12207 – Systems and software engineering - Software Life Cycle Processes*, в якому всі процеси ЖЦ ПЗ розділені на три групи (рис.5.1). У табл. 1 наведено опис основних процесів ЖЦ ПЗ ІВ відповідно до ISO

Рисунок 5.1 – Процеси ЖЦ ІС відповідно до стандарту ISO 12207

Таблиця 1 - Зміст *основних процесів* ЖЦ ПЗ ІВ відповідно до ISO 12207

|  |  |  |  |
| --- | --- | --- | --- |
| **Процес**  **(Виконання)** | **Дії** | **Вхід** | **Результат** |
| **Купівля (Замовник)** | * ініціювання; | * рішення про початок; * впровадження ІС; * результати дослідження діяльності замовника; * результати аналізу ринку ІВ/тендера; * план поставки/ розробки; * комплексний тест ІС | * техніко-економічне обґрунтування впровадження ІС; * технічне завдання на ІС; * угода на постачання / розробку; * акти приймання етапів роботи; * акт приймально- передавальних випробувань |
| * підготовка вимог заявки; * підготовка угоди; * контроль діяльності постачальника; * прийом ІС |
|  |
| **Поставки**  **(Розробник ІС)** | * ініціювання; * відповідь на замовлення; | * технічне завдання на ІС; * рішення про участь | * рішення про участь у розробці; * комерційна пропозиція / конкурсна заявка; * угода про постачання / розробки; * план управління проектом; * реалізація / коригування; * акт приймально- передавальних випробувань; |
| * підготовка угоди; | у розробці; |
| * планування виконання; * поставка ІС | * результати тендеру; * технічне завдання на ІС; * план управління проектом; * створення ІС та документації |
|  |  |

*Допоміжні процеси* призначені для підтримки виконання основних процесів, забезпечення якості проекту, організації верифікації та тестування ПЗ.

*Організаційні процеси* визначають дії і завдання замовників і розробників для управління процесами в ході проекту.

Для підтримки практичного використання стандарту ISO 12207 розроблені такі технологічні документи:

* Керівництво для ISO / IEC 12207 (ISO / IEC TR 24748-3: 2011 Systems and software engineering - Life cycle management - Part 3: Guide to the application of ISO / IEC 12207 (Software life cycle processes))
* Керівництво по використанню ISO / IEC 12207 в управлінні проектами (ISO / IEC TR 16326: 2009 Systems and software engineering - Life cycle processes - Project management).

У 2002 був опублікований стандарт на процеси життєвого циклу систем ***ISO / IEC 15288 Systems and software engineering – System life cycle processes****,* в розробці якого брали участь фахівці різних галузей: системної інженерії, програмування, управління якістю, людськими ресурсами, безпекою та ін.

Цей документ враховує практичний досвід створення систем в урядових, комерційних, військових та академічних організаціях і може бути застосований для широкого класу систем, але *його основне призначення - підтримка створення комп'ютеризованих систем.*

В даний час діє версія стандарту 2008. *У стандарті ISO / IEC 15288: 2008* в структурі ЖЦ виділені групи процесів за видами діяльності (рис.15).

Стадії створення системи, передбачені в стандарті ISO / IEC 15288: 2008, і основні результати, що мають бути досягнуті до моменту їх завершення, наведені в таблиці 2.

*Стандарти ISO / IEC 12207 та ISO / IEC 15288 мають єдину термінологію* і розроблені таким чином, щоб могли використовуватися одночасно в проекті.

Таблиця 2. - Стадії *створення систем* (ISO / IEC 15288)

|  |  |  |
| --- | --- | --- |
| №  п/п | Стадія | Опис |
| 1 | Формування концепції | Аналіз потреб, вибір концепції та проектних  Рішень |
| 2 | Розробка | Проектування системи |
| 3 | Реалізація | Створення системи |
| 4 | Експлуатація | Введення в експлуатацію і використання системи |
| 5 | Підтримка | Забезпечення функціонування системи |
| 6 | Зняття з експлуатації | Зупинка використання, демонтаж, архівування системи |

Також потрібно відзначити, що в процесі промислової розробки ПЗ обов'язково використовуються *стандарти якості серії* ***ISO 9000***.

Серія ISO 9000 - (управління якістю) включає в себе такі стандарти:

ISO 9000-1. Управління якістю і гарантії якості. Частина 1. Керівництво з вибору і використання.

ISO 9000-2. Управління якістю і гарантії якості. Частина 2. Загальне керівництво по застосуванню стандартів ISO 9001, ISO 9002 та ISO 9003. ISO 9000-3. Управління якістю і гарантії якості. Частина 3.

Керівництво по застосуванню стандарту ISO 9001 при розробці, установці і супроводі ПЗ. ISO 9000-4. Управління якістю і гарантії якості. Частина 4. Керівництво з управління надійністю програм.

*Основний стандарт ISO 9001: 2009* задає модель системи якості для процесів проектування, розробки, виробництва, установки і обслуговування (продукту, системи, послуги).

У моделі ISO 9000 лише згадуються вимоги, які повинні бути реалізовані, але не говориться, як це можна зробити. Тому для побудови повноцінної системи якості ISO, крім основної моделі ISO 9001, необхідно використовувати допоміжні галузеві і рекомендаційні стандарти.

1. Стандарти організації **IEEE – Institute of Electrical and Electronics Engineers**

У 1963 в результаті злиття Інституту радіотехніки (Institute of Radio Engineers, IRE) і Американського інституту інженерів - електриків (American Institute of Electrical Engineers, AIEE) була створена міжнародна некомерційна асоціація технічних фахівців - світовий лідер в області розробки стандартів з радіоелектроніки та електротехніки – інститут інженерів з радіоелектроніки та електротехніки IEEE (Institute of Electrical and Electronics Engineers). Дана міжнародна організація об'єднує понад 400 тисяч фахівців з 170 країн.

IEEE здійснює інформаційну та матеріальну підтримку фахівців для організації та розвитку наукової діяльності в електротехніці, електроніці, комп'ютерній техніці та інформатиці.

*Керівництво до зведення знань з програмної інженерії* ***(SWEBOK****, 2004)* містить опис 10 галузей знань:

*Software requirements* – програмні вимоги;

*Software design –* дизайн (архітектура);

*Software construction –* конструювання програмного забезпечення;

*Software testing –* тестування;

*Software maintenance – експлуатація (підтримка) програмного забезпечення;*

*Software configuration management – управління конфігураціями;*

*Software engineering management – управління з програмної інженері;*

*Software engineering process – процеси програмної інженерії;*

*Software engineering tools and methods – інструменти та методи;*

*Software quality–- якість програмного забезпечення.*

Для кожної галузі ***SWEBOK*** містить опис ключових елементів у

вигляді *підобластей (subareas*).

Для кожної підобласті приведена декомпозиція в *вигляді списку тем*

*(topics)* з їх описом.

**Лекція 6:** Стандарт зрілості компанії**-**розробника програмного забезпечення **CMM - Capability Maturity Model**

Говорячи про стандартизацію процесів підприємства потрібно розглянути модель зрілості технологічних процесів організації Capability Maturity Model (CMM), яка розроблена Інститутом інженерів програмного забезпечення (Software Engineering Institute, SEI) і корпорацією Mitre під керівництвом Воттса Хамфрі (Watts Humphrey).

*Методологія* ***CMM*** розроблялася і розвивалася в США як засіб, що дозволяє вибирати кращих виробників ПЗ для виконання держзамовлень в першу чергу міністерства оборони. Для цього були розроблені критерії оцінки зрілості ключових процесів компанії і певний набір дій, необхідних для їх подальшого вдосконалення.

У підсумку методологія виявилася надзвичайно корисною для більшості компаній, які прагнуть якісно поліпшити існуючі процеси проектування, розробки, тестування програмних засобів і звести управління ними в зрозумілих і легко реалізованих алгоритмів і технологій, описаних в єдиному стандарті. Надалі ця модель переросла в *методологію підвищення якості процесів підприємства Capability Maturity Model Integration CMMI.*

Застосування CMMI дозволяє поставити розробку ПЗ на промислову основу, підвищити керованість ключовими процесами і виробничу культуру в цілому, гарантувати якісну роботу та виконання проектів в строк.

Основою для створення CMM стало базове положення про те, що фундаментальна проблема "кризи" процесу розробки якісного ПЗ полягає не у відсутності нових методів і засобів розробки, а в нездатності компанії організувати технологічні процеси і керувати ними.

*Для оцінки ступеня готовності підприємства розробляти якісний програмний продукт СММ використовує ключове поняття зрілість організації* ***(Maturity*** *)*.

*Незрілою вважається організація****,*** *в якій****:***

* + відсутнє довгострокове і проектне планування;
  + процес розробки програмного забезпечення та його ключові моменти не ідентифіковані, реалізація процесу залежить від поточних умов, конкретних менеджерів і виконавців;
  + методи і процедури не стандартизовані і не задокументовані;
  + результат не визначений реальними критеріями, які встановлюються запланованими показниками із застосуванням стандартних технологій і розроблених метрик;
  + процес вироблення рішення відбувається стихійно, на грані мистецтва. У цьому випадку велика ймовірність появи несподіваних проблем, перевищення бюджету або невиконання термінів здачі проекту. У такій компанії, як правило, менеджери і розробники не управляють процесами - вони змушені займатися поточними і спонтанними проблемами.

*Основні ознаки зрілої організації****:***

* в компанії чітко визначені і документовані процедури управління вимогами, планування проектної діяльності, управління конфігурацією, створення і тестування програмних продуктів, відпрацьовані механізми управління проектами;
* ці процедури постійно уточнюються і удосконалюються;
* оцінки часу, складності та вартості робіт ґрунтуються на накопиченому досвіді, розроблених метриках і кількісних показниках, що робить їх досить точними;
* актуалізовані зовнішні і створені внутрішні стандарти на ключові процеси та процедури;
* існують обов'язкові для всіх правила оформлення методологічної і програмної документації та документації користувача;
* технології незначно змінюються від проекту до проекту, на основі стабільних і перевірених підходів та методик максимально використовуються напрацьовані в попередніх проектах організаційний і виробничий досвід, програмні модулі, бібліотеки програмних засобів;
* активно апробуються і впроваджуються нові технології, проводиться оцінка їх ефективності.

*СММ визначає п****'****ять рівнів технологічної зрілості компанії* (рис. 6.1), за якими замовники можуть оцінювати потенційних претендентів на підпис контракту, а розробники - удосконалювати процеси створення ПЗ.

Кожен з рівнів, крім першого, складається з декількох ключових областей процесу (Key Process Area), що містять цілі (Goal), зобов'язання щодо виконання (Commitment to Perform), можливість виконання (Ability to Perform), дії, що виконуються, (Activity Performed), їх вимір і аналіз (Measurement and Analysis) і перевірку впровадження (Verifying Implementation).

Таким чином, *СММ фактично є комплексом вимог до ключових параметрів ефективного стандартного процесу розробки ПЗ і засобом його постійного поліпшення.* Виконання цих вимог збільшує ймовірність досягнення підприємством поставлених цілей в галузі якості.

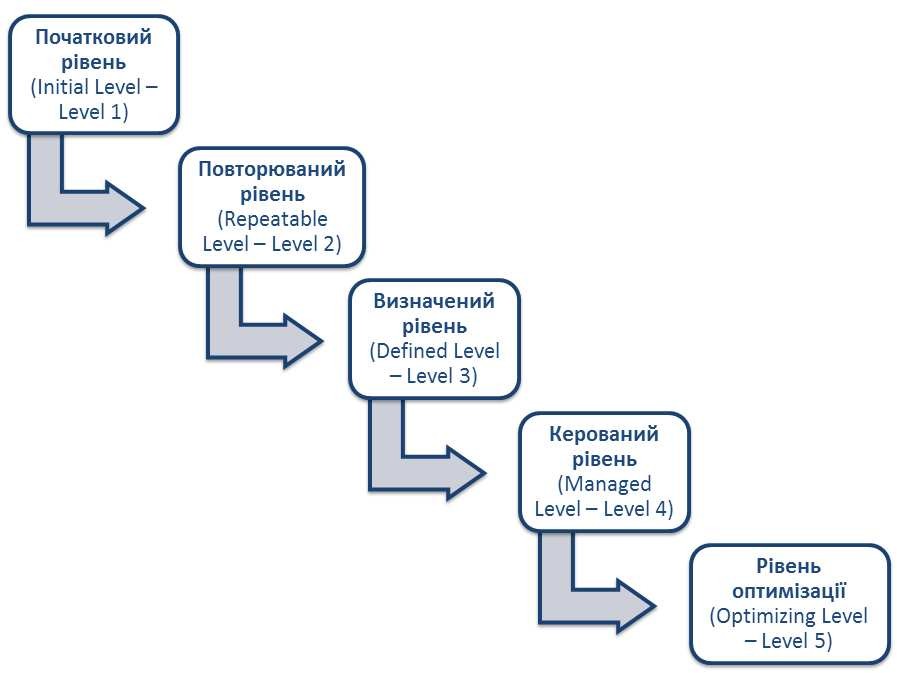


Рисунок 6.1 – Рівні зрілості компанії за стандартом СММ

*Початковий рівень (Initial Level - Level 1)*

На даному рівні компанія може отримати замовлення, розробити і передати замовнику програмний продукт. Стабільність розробок відсутня. Лише деякі процеси визначені, результат повністю залежить від зусиль окремих співробітників. Успіх одного проекту не гарантує успішності наступного. До цієї категорії можна віднести будь-яку компанію, яка хоч якось виконує взяті на себе зобов'язання. Ключові області процесу цього рівня не зафіксовано.

*Повторюваний рівень (Repeatable Level - Level 2)*

Цьому рівню відповідають підприємства, що володіють певними технологіями управління та розробки. Управління вимогами та планування в більшості випадків ґрунтуються на розробленій документованій політиці та накопиченому досвіді. Встановлено та введено в повсякденну практику базові показники для оцінки параметрів проекту. Менеджери відстежують виконання робіт і контролюють тимчасові і виробничі витрати.

У компанії розроблені деякі внутрішні стандарти і організовані спеціальні групи *перевірки якості QA*. Зміни версій кінцевого програмного продукту і створених проміжних програмних засобів відслідковуються в системі управління конфігурацією. Є необхідна дисципліна дотримання встановлених правил. Ефективні методики та процеси впроваджуються, що забезпечує можливість повторення успіху попередніх проектів в тій же прикладній галузі. Ключові галузі процесу розробки ПЗ цього рівня наведені на рисунку 6.2.

*Визначний рівень (Defined Level - Level 3)*

Рівень характеризується деталізованим методологічним підходом до управління (описані і закріплені в документованій політиці типові дії, що необхідні для багаторазового повторення: ролі і відповідальність учасників, стандартні процедури та операції, порядок дій, кількісні показники і метрики процесів, формати документів та ін.).

Для створення і підтримки методологій в актуальному стані в організації підготовлена і постійно функціонує спеціальна група (рис. 6.2)



Рис. 6.2 – Ключові області повторюваного і визначного рівнів зрілості компанії

Компанія регулярно проводить тренінги для підвищення професійного рівня своїх співробітників. Починаючи з цього рівня, організація практично перестає залежати від особистісних якостей конкретних розробників і не має тенденції опускатися на більш низькі рівні. Ця незалежність обумовлена продуманим механізмом постановки завдань, планування заходів, виконання операцій і контролю виконання. Управлінські та інженерні процеси документовані, стандартизовані і інтегровані в уніфіковану для всієї організації технологію створення ПЗ. Кожен проект використовує затверджену версію цієї технології, адаптовану до особливостей поточного проекту. Ключові галузі процесу розробки ПЗ цього рівня наведені на рисунку 6.2.

*Керований рівень (Managed Level – Level 4)*

Рівень, на якому розроблені і закріплені у відповідних нормативних документах кількісні показники якостей.

Більш високий рівень управління проектами досягається за рахунок зменшення відхилень різних показників проекту від запланованих. При цьому тенденції зміни продуктивності процесу можна відокремити від випадкових варіацій на підставі статистичної обробки результатів вимірювань в процесах. Ключові області процесу розробки ПЗ цього рівня наведені на рисунку 3.4.

*Рівень оптимізації (Optimizing Level – Level 5)*

Для цього рівня заходи щодо вдосконалення розраховані не тільки на існуючі процеси, а й на впровадження, використання нових технологій і оцінку їх ефективності. Основним завданням всієї організації на цьому рівні є постійне вдосконалення існуючих процесів, в ідеалі спрямоване на запобігання відомої помилки або дефекту і попередження можливих. Застосовується механізм повторного використання компонентів від проекту до проекту (шаблони звітів, формати вимог, процедури і стандартні операції, бібліотеки модулів програмних засобів). Ключові галузі процесу розробки ПЗ цього рівня наведені на рисунку 6.3.



Рисунок 6.3 – Ключові галузі керованого рівня зрілості і рівня оптимізації компанії

*СММ визначає такий мінімальний набір вимог****:*** *реалізувати 18*

*ключових галузей процесу розробки ПЗ, що містять 52 мети, 28 зобов'язань компанії, 70 можливостей виконання (гарантій компанії) і 150 ключових практик.*

В результаті аудиту та атестації компанії присвоюється певний рівень, за результатами наступних аудитів рівень може підвищуватися або знижуватися. Кожен наступний рівень в обов'язковому порядку включає в себе всі ключові характеристики попередніх. У зв'язку з цим сертифікація компанії в одному з рівнів передбачає безумовне виконання всіх вимог більш низьких рівнів.

*До переваг моделі* ***CMM*** *відноситься* те, що вона орієнтована на організації, які займаються розробкою програмного забезпечення. У даній моделі вдалося більш детально визначити вимоги, які специфічні для процесів, що пов'язані з розробкою ПЗ. З цієї причини в CMM наведені не тільки вимоги до процесів організації, а й приклади реалізації таких вимог.

*Основний недолік* ***CMM*** полягає в тому, що модель не авторизована як стандарт ні міжнародними, ні національними органами стандартизації. Втім, CMM давно стала промисловим стандартом. До недоліків моделі також необхідно віднести великі зовнішні накладні витрати на приведення процесів компанії у відповідність до моделі СММ, ніж при використанні моделей Міжнародного стандарту ISO 9000.

1. Стандарт **SPICE**

*Стандарт* ***SPICE*** успадкував багато рис більш ранніх стандартів, у тому числі і вже згадуваних ISO 9001 і СММ. Найбільше SPICE нагадує СММ. Точно так само, як і в СММ, основним завданням організації є постійне поліпшення процесу розробки можливостей (в SPICE визначено 6 різних рівнів), але ці рівні застосовуються не тільки до організації в цілому, а й до окремо взятих процесів.

В основі стандарту лежить *оцінка процесів*. Ця оцінка виконується шляхом порівняння процесу розробки програмного забезпечення, що існує в даній організації, з описаною в стандарті моделлю. Аналіз результатів, що отримані на цьому етапі, допомагає визначити сильні і слабкі сторони процесу, а також внутрішні ризики, що властиві даному процесу. Це допомагає оцінити ефективність процесів, визначити причини погіршення якості та пов'язані з цим витрати в часі або вартості. Потім виконується визначення можливостей процесу, тобто можливостей його поліпшення. В результаті в організації може з'явитися розуміння необхідності поліпшення того чи іншого процесу. До цього моменту цілі вдосконалення процесу вже чітко сформульовані і залишається тільки технічна реалізація поставлених завдань. Після цього весь цикл робіт починається спочатку.

Безумовно, вдосконалення процесів життєвого циклу програмного забезпечення абсолютно необхідне. Проте слід мати на увазі, що побудова «більш зрілого» процесу розробки не обов'язково забезпечує створення більш якісного програмного забезпечення. Це хоча і пов'язані, але абсолютно різні процеси.

Використання формальних моделей і методів дозволяє створювати зрозумілі, несуперечливі специфікації на програмне забезпечення, що розробляється. Звичайно, впровадження таких методів має сенс, хоча воно дуже дороге і трудомістке, а можливості його застосування досить обмежені. Основна ж проблема - проблема складності програмного забезпечення з удосконаленням процесів розробки поки не вирішена.

Створення програмного забезпечення як і раніше висуває підвищені вимоги до кваліфікації тих, хто цим займається: проектувальникам програмного забезпечення і безпосередньо програмістам.