

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ АВТОМОБІЛЬНО-ДОРОЖНИЙ
УНІВЕРСИТЕТ

ОПОРНИЙ КОНСПЕКТ ЛЕКЦІЙ

з дисципліни «Стандартизація та сертифікація програмного забезпечення», для студентів галузі знань 12 «Інформаційні технології» спеціальностей 122 «Комп'ютерні науки», 121 «Інженерія програмного забезпечення» освітньо-кваліфікаційного рівня «бакалавр»

Затверджено методичною радою
механічного факультету, протокол № 10 від 10 «червня» 2022 р.

Укладач: к.т.н., доц. Букрєєва О. С.

Харків
ХНАДУ
2022

ЗМІСТ

Вступ	2
Лекція 1 Основні поняття стандартизації	3
Лекція 2 Стандартизація програмних засобів	10
Лекція 3 Загальні засади оцінки відповідності	15
Лекція 4 Правове регулювання у галузі комп'ютерних технологій	20
Лекція 5 Основні поняття теорії надійності програмних засобів. Моделі надійності	24
Лекція 6 Вимоги до якості систем і програмних засобів та її оцінювання	31
Лекція 7 Моделі якості	35
Лекція 8 Метрики якості	39
Лекція 9 Процеси життєвого циклу систем	46
Лекція 10 Процеси життєвого циклу продукту	57
Лекція 11 Моделі життєвого циклу	65
Лекція 12 Керування ризиками	73
Лекція 13 Вимоги до супроводу програмного забезпечення	79
Лекція 14 Поняття і визначення в тестуванні	87
Лекція 15 Тестові процеси	92
Лекція 16 Тестова документація	99

ВСТУП

Цей конспект лекцій призначений для студентів третього курсу галузі знань 12 «Інформаційні технології» спеціальностей 122 «Комп'ютерні науки» та 121 «Інженерія програмного забезпечення» з метою надання допомоги вивченні дисципліни «Стандартизація та сертифікація програмного забезпечення». Її **метою** є надання студентам системи теоретичних і практичних знань у галузі стандартизації та сертифікації програмного забезпечення, що дозволить використовувати їх у професійній діяльності. Після завершення її вивчення студенти будуть:

знати: основні поняття стандартизації, види нормативних документів, поняття сертифікації та підтвердження відповідності, принципи, методи, правила, схеми сертифікації продукції, процесів та послуг в системі сертифікації, основні поняття теорії надійності програмних засобів, вимоги до якості систем і програмних засобів та її оцінювання, модель оцінювання процесу для тестування програмного забезпечення;

вміти: визначити вимоги до конкретного програмного продукту за нормативними документами, визначити відповідність продукції за знаком відповідності, підготувати комплект документів, необхідний для сертифікації програмного забезпечення та процесів його виробництва, визначити якість програмного продукту за метриками, організувати сертифікаційні тестування програмних продуктів, самостійно розроблювати різні види документів для сертифікації програмного забезпечення.

Конспект лекцій складено згідно з програмою дисципліни, стандартами вищої освіти з підготовки бакалаврів вказаних спеціальностей, освітньою програмою та навчальним планом.

ЛЕКЦІЯ 1

ОСНОВНІ ПОНЯТТЯ СТАНДАРТИЗАЦІЇ

Мета – знати історичний розвиток стандартизації, основні терміни та визначення, об’єкти, суб’єкти, функції та принципи стандартизації; вміти визначати вид нормативного документу за його позначенням.

ПЛАН

1. Історичний розвиток стандартизації
2. Основні терміни та визначення
3. Об’єкти, суб’єкти, функції та принципи стандартизації
4. Види нормативних документів

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Правила проведення робіт з національної стандартизації
2. Міжнародне співробітництво у галузі стандартизації

ЛІТЕРАТУРА

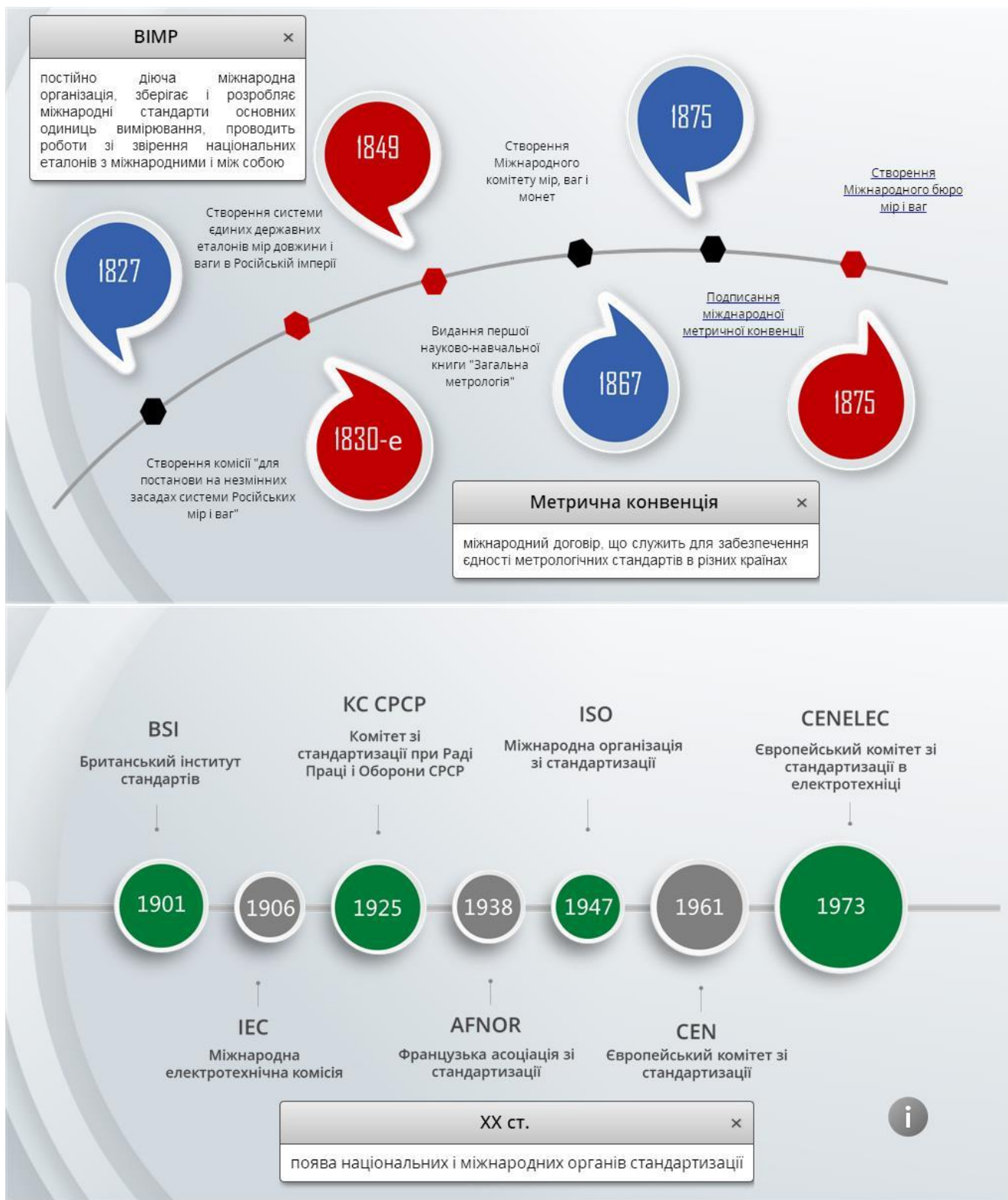
1. Янушкевич Д.А. Національна та міжнародна стандартизація / Д.А. Янушкевич, О.А. Коваль. – Х.: ХНАДУ, 2010.- 237 с.
2. Саранча Г.А. Метрологія, стандартизація, відповідність, акредитація та управління якістю: Підручник. – К.: Центр навчальної літератури, 2006. – 672 с.
2. Букреєва О. С. Основи стандартизації та оцінки відповідності : електрон. навч. посіб. у схемах і табл. [Електронний ресурс] / О. С. Букреєва, І. В. Рибалко ; М-во освіти і науки України, Харків. нац. автомоб.-дор. ун-т. – Харків, 2019. – 76 с. URL: https://dspace.khadi.kharkov.ua/dspace/bitstream/123456789/2532/3/Bykreeva_Rybalko_osnovy_stand_2019.pdf.
3. Сидорчук О.В. Стандартизація та сертифікація техніки і обладнання /О.В. Сидорчук, Ковалишин О.В., Городецький І.М. – Львів: Львівський ДАУ, 2007. – 189 с.
4. ДСТУ 1.1:2015 Національна стандартизація. Стандартизація та суміжні види діяльності. Словник термінів (ISO/IEC Guide 2:2004, MOD)
5. ДСТУ 1.2:2015 Національна стандартизація. Правила проведення робіт з національної стандартизації
6. Закон України від 05.06.2014 № 1315-VII «Про стандартизацію»

КОНТРОЛЬНІ ПИТАННЯ

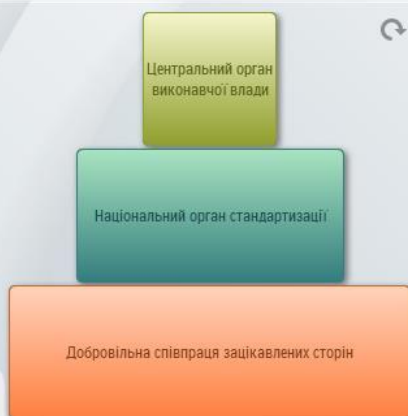
1. Дайте визначення поняттям «стандартизація», «сертифікація».
2. Який орган є найстарішим національним органом стандартизації?
3. Які є рівні стандартизації?
4. На яких принципах базується державна політика у галузі стандартизації?
5. Які є види нормативних документів?

ОПОРНИЙ КОНСПЕКТ





Органи & Документи	1992	1994	1996	1998	2000	2002	2004	2006	2008	2010	2012	2014	2016	2018
Органи														
•Державний комітет стандартизації, метрології та сертифікації України														
•Державний Комітет України з питань технічного регулювання та споживчої політики														
•Державна інспекція України з питань захисту прав споживачів														
•Департамент технічного регулювання														
Документи														
•Декрет Кабінету Міністрів України «Про стандартизацію і сертифікацію»														
•Закон України «Про стандартизацію»														
•Закон України «Про технічні регламенти та оцінку відповідності»														



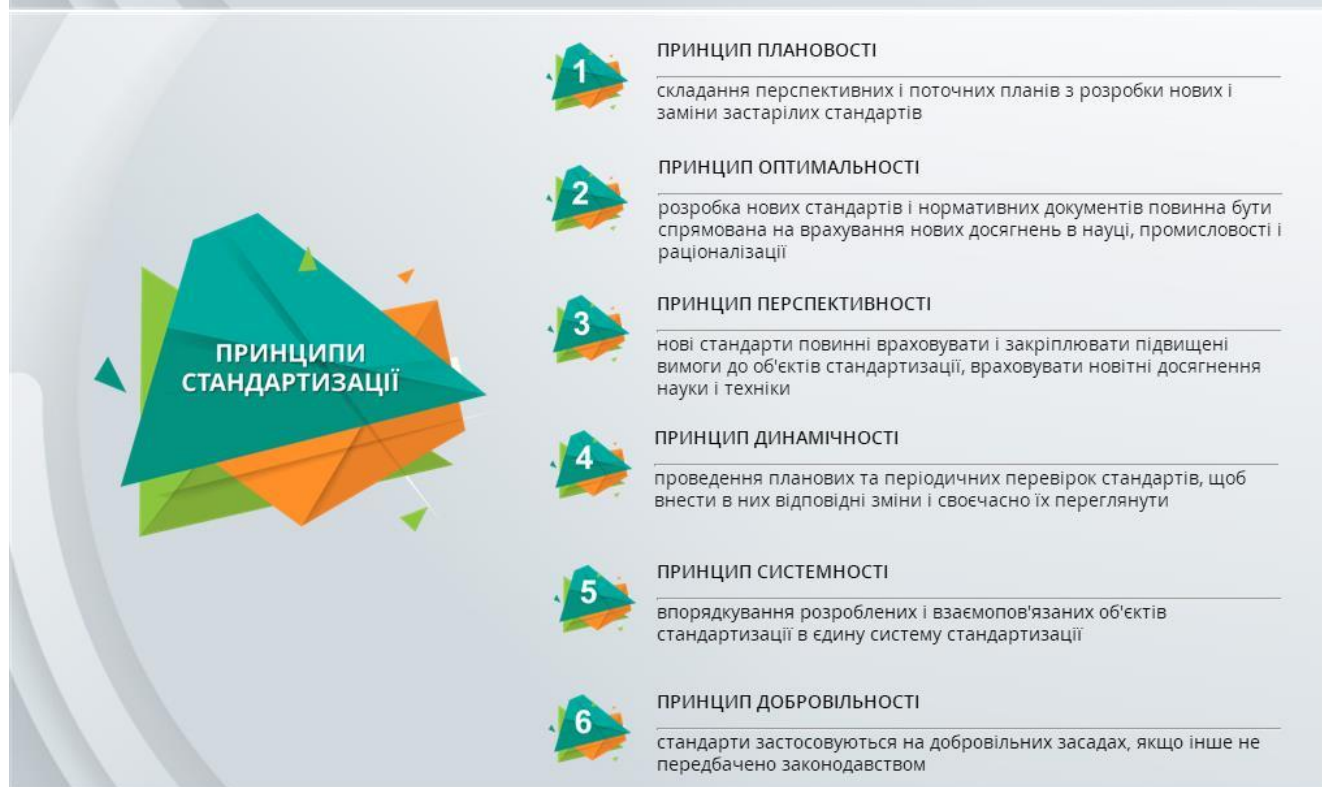
○ Департамент технічного регулювання
 ○ УкрНДНЦ
 ○ Технічний комітет

проводиться на міжнародному рівні, участь відкрито для відповідних органів усіх країн

проводиться на відповідному регіональному рівні, участь відкрито для національних органів країн певного географічного або економічного простору

проводиться на рівні однієї країни





Функції стандартизації



Впорядку-
вання

Охоронна
(соціальна)

Ресурсо-
збережна

Комуніка-
тивна

Цивіліза-
ційна

Інформа-
ційна

Нормо-
творчості

подолання нерозумного різноманіття об'єктів

організаційно-методичні та загальнотехнічні

продукція, призначена для використання в різних видах економічної діяльності, для державних закупівель та широкого вжитку

системи і господарські об'єкти, які мають важливе значення, і їх складові, зокрема транспорт, зв'язок, енергосистема, використання природних ресурсів



потреби оборони, мобілізаційної готовності та державної безпеки

будівельні матеріали, процеси, типові деталі та будинки, системи функціонального забезпечення будинків, складні будівельні споруди і методи контролю в будівництві

вимоги щодо захисту прав споживачів, охорони праці та навколишнього природного середовища, ергономіки,

організаційно-методичні та загальнотехнічні об'єкти стандартизації



організація робіт зі стандартизації

1

2

3

4

5

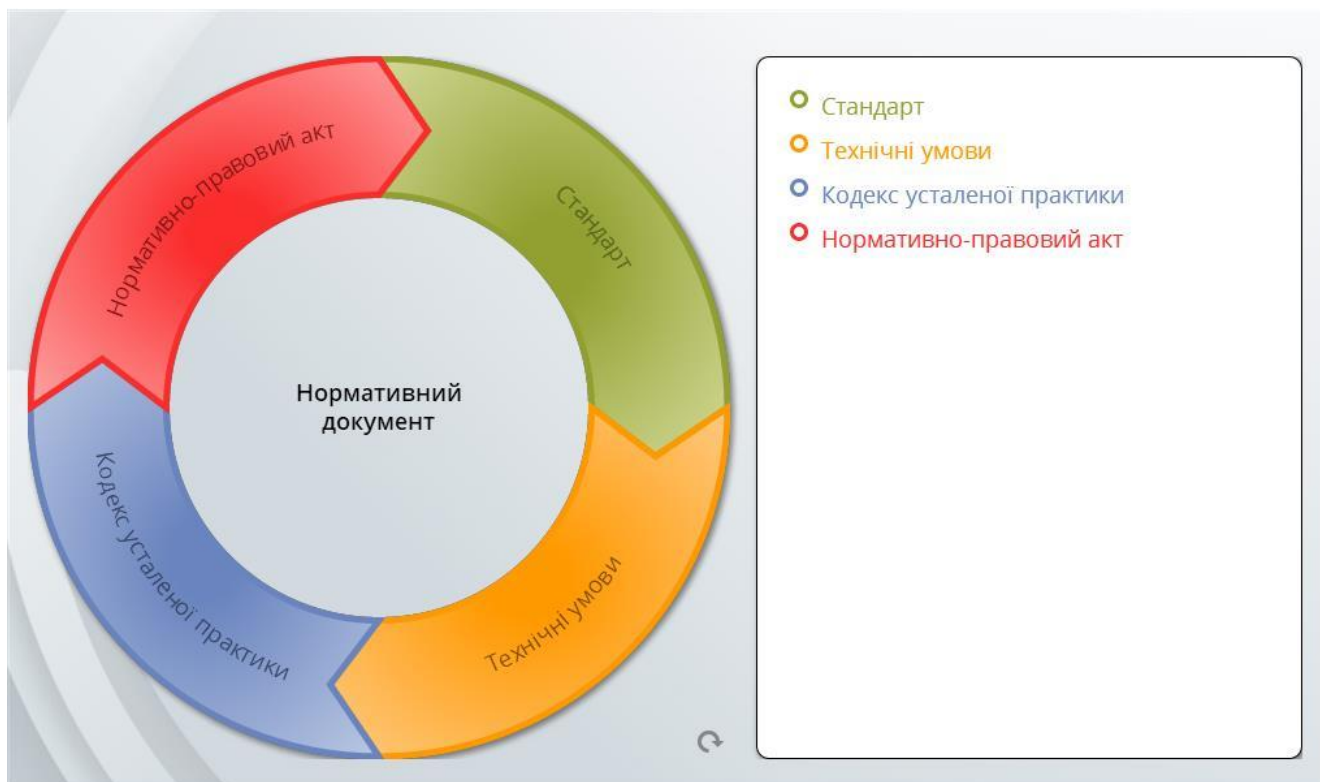
6

7

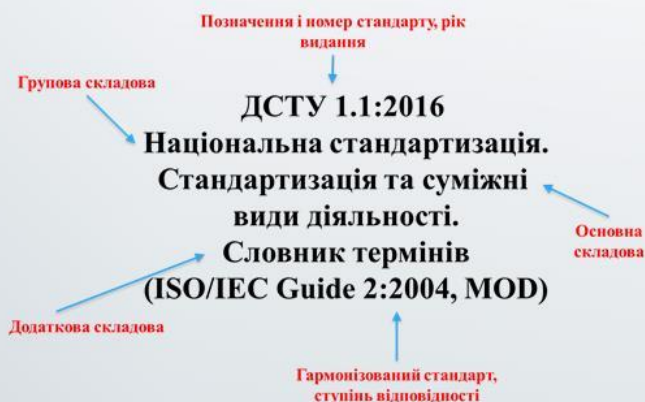
8

9

10



Міжнародний	ISO, IEC, ISO/IEC
Міжрегіональний	ГОСТ
Європейський	EN
Національний	ДСТУ
Галузевий	ОСТ, ГСТУ
Стандарт підприємства	СОУ, СПП, СТО
Пробний	ДСТУ-П
Керівництво	Р
Настанова	ДСТУ-Н
Методичні вказівки	МИ
Класифікатор	ДК
Звіт	ДСТУ-ЗТ
Технічні умови	ТУ, ТУ У
Керівний НД	КНД
Внедренные	
Міжнародний	ДСТУ IEC 60269-1-2001
Міжрегіональний	ДСТУ ГОСТ 30712:2003



ЛЕКЦІЯ 2

СТАНДАРТИЗАЦІЯ ПРОГРАМНИХ ЗАСОБІВ

Мета – знати міжнародний досвід стандартизації програмного забезпечення, профілі стандартів з програмного забезпечення, способи його документування.

ПЛАН

1. Міжнародний досвід стандартизації програмного забезпечення
2. Профілі стандартів з програмного забезпечення
3. Серія стандартів ISO 25000
4. Документування програмних засобів

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Нормативно-методичне забезпечення створення програмного забезпечення
2. Стандарти підприємства

ЛІТЕРАТУРА

1. Липаев В. В. Сертификация программных средств: Учебник / В. В. Липаев. – М.: Синтег, 2009. – 336 с.
2. Глухова Л. А. Конспект лекций по курсу «Стандартизация и сертификация программного обеспечения» / Л. А. Глухова. – Минск: БГУИР, 2004. – 80 с.

КОНТРОЛЬНІ ПИТАННЯ

1. Які є види профілів стандартів з програмного забезпечення?
2. З яких розділів складається серія стандартів ISO 25000?
3. Яка мета документування програмних засобів?
4. Які є види програмних документів?
5. Які інформаційні елементи можуть бути описані у програмних документах?

ОПОРНИЙ КОНСПЕКТ

ОСОБЛИВОСТІ СТАНУ ТА РОЗВИТКУ СТАНДАРТИЗАЦІЇ В ГАЛУЗІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

розроблено кілька сотень міжнародних та національних стандартів; проте вони не повністю і не рівномірно покривають потреби в стандартизації об'єктів і процесів створення та застосування складних систем і їх компонентів

01

в області систем стандартами регламентовані найбільш прості об'єкти і процеси (телекомунікації, програмування, документування і т.п.)

02

велика тривалість розробки, погодження та затвердження міжнародних і національних стандартів (3 - 5 років); це призводить до відставання вимог і рекомендацій стандартів від сучасного стану техніки, потреб практики і технологій створення складних систем

03

складні процеси життєвого циклу програмного забезпечення (системний аналіз і проектування, інтеграція компонентів, випробування, сертифікація) майже не описані стандартами через труднощі їх формалізації та уніфікації

04

стандарты не завжди враховують побудову програмного забезпечення як відкритих систем і не забезпечують їх розширюваність, переносимість між різними платформами, можливість взаємодії з іншими інформаційними системами

05

прогалини і затримки в підготовці і виданні міжнародних і національних стандартів призводять до створення і застосування численних нормативних і методичних документів галузевого або фірмового рівня

06

профілі, які регламентують архітектуру і структуру програмних засобів і їх компонентів (функції, інтерфейси, протоколи взаємодії, формати даних і т.п.)

4

профілі конкретного програмного засобу, діють в межах проекту і є частиною проектної документації

1

профілі, які регламентують процеси і системи забезпечення якості проектування, розробки, застосування, супроводу і розвитку програмних засобів і їх компонентів

3

профілі для вирішення певного класу прикладних задач, поширюються на всі програмні засоби даного класу, затверджуються як стандарти підприємств, галузеві або державні стандарти

2



- 01 **Institute of Electrical and Electronic Engineers**
- 02 **International Standards Organization**
- 03 **Electronic Industry Association**
- 04 **International Electrotechnical Commission**
- 05 **American National Standards Institute**
- 06 **Software Engineering Institute**

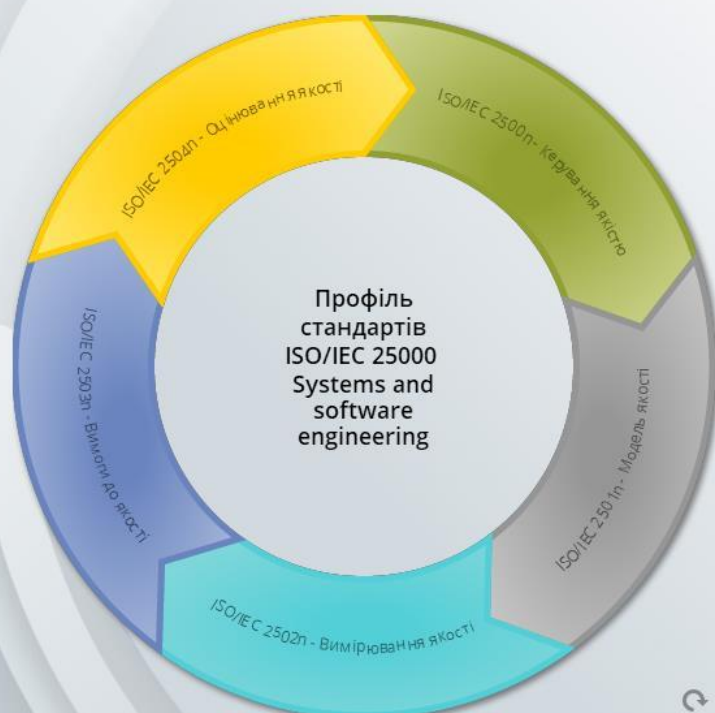
Стандарти IEEE

Описують структуру процесів розробки і супроводу, а також процесів, пов'язаних з ними, визначають основні види діяльності, які виконуються в рамках цих процесів, і документи, які потрібні на вході і виникають на виході цих видів діяльності.

ОРГАНІЗАЦІЇ, ЯКІ РОЗРОБЛЯЮТЬ СТАНДАРТИ В ГАЛУЗІ ПЗ

Стандарти ISO

Визначають структуру процесів життєвого циклу ПЗ, правила оцінки процесів життєвого циклу ПЗ і їх можливостей, спираються на модель технологічної зрілості CMMI (Capability Maturity Model Integrated - не висуває точних вимог, встановлює мінімальні критерії для раціонального процесу в багатьох середовищах розробки ПЗ) і орієнтовані на оцінку можливості поліпшення процесів.



- ISO/IEC 2500n - Керування якістю
- ISO/IEC 2501n - Модель якості
- ISO/IEC 2502n - Вимірювання якості
- ISO/IEC 2503n - Вимоги до якості
- ISO/IEC 2504n - Оцінювання якості

ДОКУМЕНТУВАННЯ ПРОГРАМНИХ ЗАСОБІВ



описати і закріпити інформацію про програмне забезпечення протягом його життєвого циклу



сприяти забезпеченню практичності і придатності до супроводу програмного забезпечення



допомогти контролювати процес випуску програмного забезпечення



надати інформацію про програмне забезпечення тим, хто її потребує

Рівні інформаційних елементів



потрібні в підсумкових документах, на ранніх етапах життєвого циклу або для простих програмних засобів, які потребують детальних документів

2

3

Складання профілю документації



визначити, які документи необхідні протягом життєвого циклу

крок 1

- 11 Визначення проблеми
- 12 Цілі
- 13 Обґрунтування пропозицій
- 14 Посилання документа
- 15 Обмеження
- 16 Мова
- 17 Контактна інформація

Системні
вимоги

Опис
системи

- 21 Ідентифікація системи
- 22 Детальні вимоги
- 23 Компоненти ПЗ
- 24 Тестування
- 25 Інсталяція
- 26 Людські ресурси

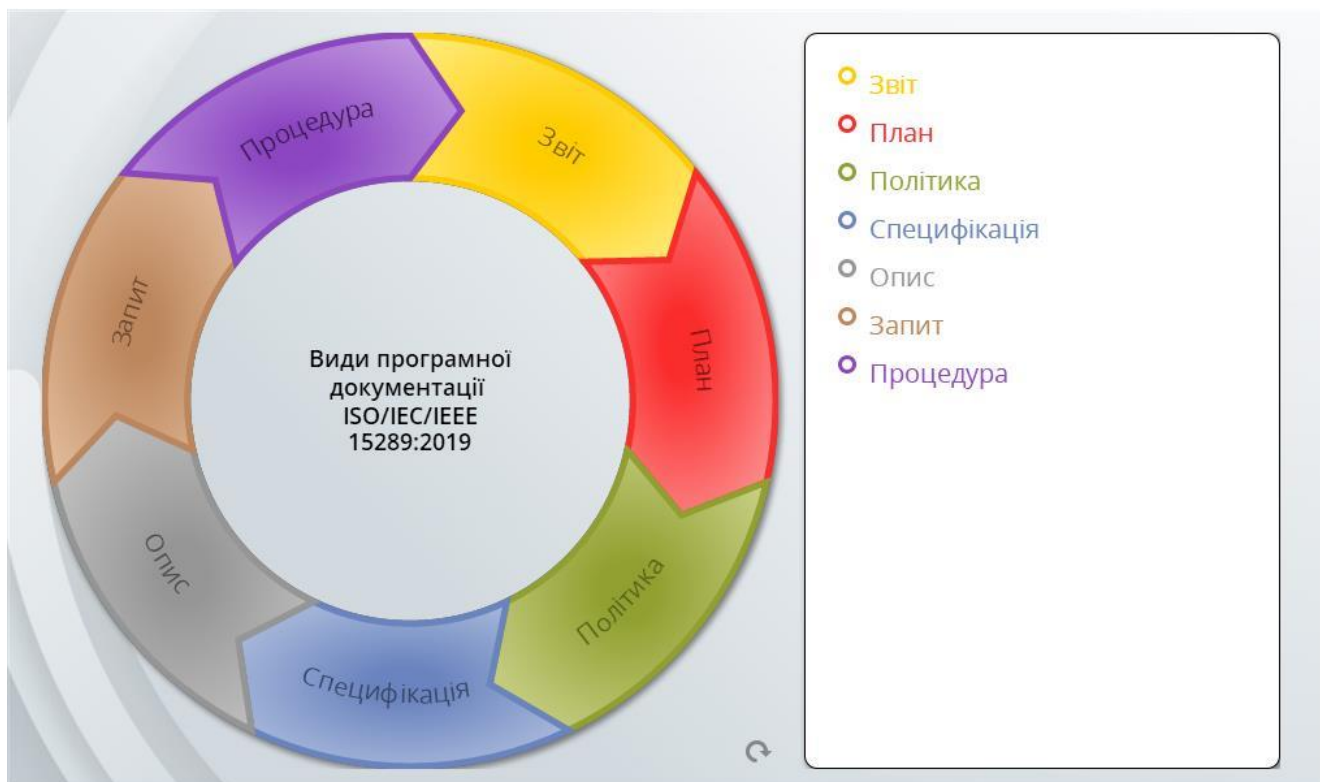
ІНФОРМАЦІЙНІ ЕЛЕМЕНТИ
ПРОФІЛЮ ДОКУМЕНТАЦІЇ

- 41 Рішення
- 42 Обґрунтування
- 43 Відповідальність

Висновки /
рекомендації

Показники
оцінювання

- 31 Витрати / вигоди
- 32 Досвід інсталяції
- 33 Сильні / слабкі сторони
- 34 Критерії / методи оцінки
- 35 Результати оцінювання



ПРИКЛАД ПРОФІЛЮ ДОКУМЕНТАЦІЇ

Інформаційний елемент			Документи		
Номер	Назва	Рівень	Технічне завдання	Керівництво користувача	Тест-план
	1 Системні вимоги				
11	Визначення проблеми	С	+	+	+
		О	+		+
		Н	+		
12	Цілі	С	+	+	+
		О	+		+
		Н	+		+
13	Обґрунтування пропозицій	С	+		
		О	+		
		Н	+		
13	Посилання документа	С	+	+	+
		О			
		Н			
і так далі					

ЛЕКЦІЯ 3

ЗАГАЛЬНІ ЗАСАДИ ОЦІНКИ ВІДПОВІДНОСТІ

Мета – знати історичний розвиток сертифікації, основні терміни та визначення, будову та призначення технічних регламентів, вид, форму та зміст документів, необхідних для сертифікації продукції; вміти визначати знаки відповідності та їх справжність, заповнювати заявку на сертифікацію продукції.

ПЛАН

1. Основні терміни та визначення
2. Знаки відповідності
3. Модулі оцінки відповідності
4. Схеми сертифікації
5. Порядок оцінки відповідності та сертифікації продукції

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Історичний розвиток сертифікації
2. Європейські правила технічного регулювання

ЛІТЕРАТУРА

1. Янушкевич Д. А. Міжнародна та національна стандартизація: навч.-метод. посіб. / Д. А. Янушкевич, О. А. Коваль; Харків. нац. автомоб.-дор. ун-т. – Х.: ХНАДУ, 2010. – 295 с.
2. Саранча Г.А. Метрологія, стандартизація, відповідність, акредитація та управління якістю: Підручник. – К.: Центр навчальної літератури, 2006. – 672 с.
3. Букреева О. С. Основи стандартизації та оцінки відповідності : електрон. навч. посіб. у схемах і табл. [Електронний ресурс] / О. С. Букреева, І. В. Рибалко ; М-во освіти і науки України, Харків. нац. автомоб.-дор. ун-т. – Харків, 2019. – 76 с. URL: https://dspace.khadi.kharkov.ua/dspace/bitstream/123456789/2532/3/Bykreeva_Rybalko_osnovy_stand_2019.pdf.
4. Сидорчук О.В. Стандартизація та сертифікація техніки і обладнання / О.В. Сидорчук, О.В. Ковалишин, І. М. Городецький. – Львів: Львівський ДАУ, 2007. – 189 с.
5. ISO/IEC 17000:2004 Conformity assessment — Vocabulary and general principles
6. ISO/IEC 17067:2013 Conformity assessment — Fundamentals of product certification and guidelines for product certification schemes
7. Закон України від 15.01.2015 № 124-VIII «Про технічні регламенти та оцінку відповідності»

КОНТРОЛЬНІ ПИТАННЯ

1. Хто є першою, другою та третьою стороною у процесі сертифікації?
2. Які модулі оцінки відповідності використовують на етапах проектування продукції?
3. За якою схемою сертифікують послуги?
4. Якими можуть бути результати технічного нагляду за сертифікованою продукцією?
5. Як виглядає український зак відповідності?

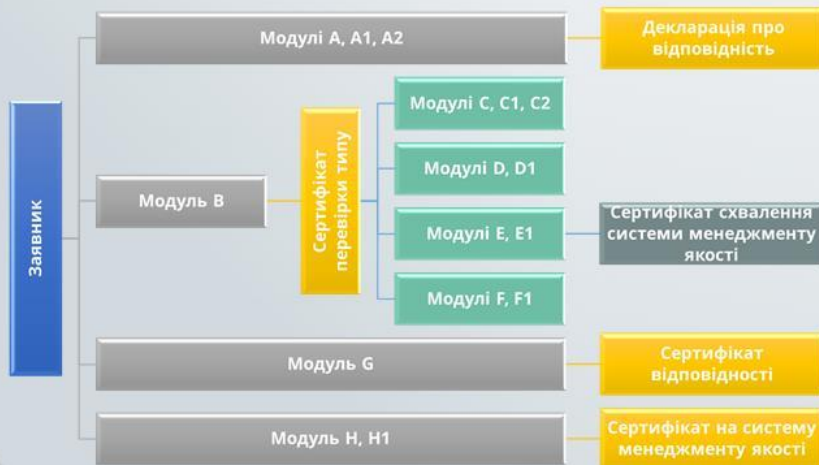
ОПОРНИЙ КОНСПЕКТ



Модулі оцінки відповідності



A = внутрішній контроль виробництва



Порядок оцінки відповідності продукції вимогам технічних регламентів



Для початку оцінки відповідності заявник звертається в будь-який зручний уповноважений орган оцінки відповідності для отримання бланка заявки і пояснень щодо її заповнення. При заповненні заявки представник органу надає пояснення щодо процедури, прав і обов'язків заявників, інформації про сплату, інформації про процедури для вирішення скарг і апеляцій і будь-яких інших питань з боку замовника. Заявник повинен подати заявку, оформлену відповідно до форми на паперовому носії, підписану вищим керівництвом і завірену печаткою заявника.



Елементи схем добровільної сертифікації	Схеми добровільної сертифікації						
	1а	16	2	3	4	5	6
1) Відбирання зразків	+	+	+	+	+	+	
2) Визначення характеристик продукції для: випробувань технічного нагляду оцінювання проекту оцінювання послуг	+	+	+	+	+	+	+
3) Оцінювання стосовно: результатів випробувань або інспектування зразків, відібраних на відкритому ринку результатів випробувань або інспектування зразків, відібраних на виробництві аудиту системи управління якістю в комбінації з випадковими випробуваннями або інспектуванням оцінювання процесів виробництва продукції або надання послуг	+	+	+	+	+	+	+
4) Розгляд результатів (випробувань, оцінки відповідності)	+	+	+	+	+	+	+
5) Рішення про сертифікацію	+	+	+	+	+	+	+
6) Ліцензування		+	+	+	+	+	+

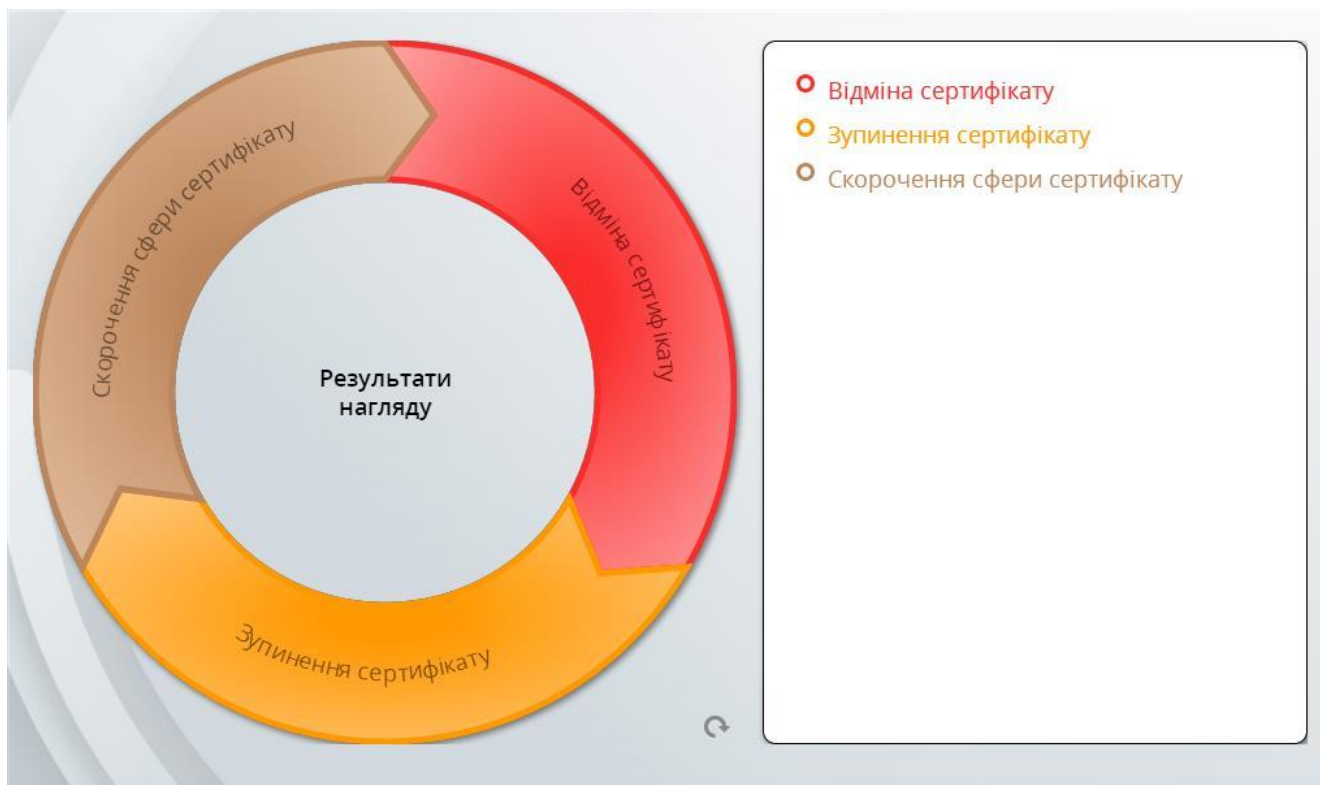
Схема	Процедури				Можливе застосування для оцінки відповідності за модулями
	Випробування	Обстеження	Оцінювання системи якості	Нагляд	
1а	+				B, G, F1
16, 2	+			+	A2
3, 4	+	+		+	C2
5	+		+	+	D, E, H1
6	Сертифікація послуг				



ПОРЯДОК ДОБРОВОЛЬНОЇ СЕРТИФІКАЦІЇ

✓	подання та розгляд заявки на сертифікацію та опитувальної анкети
✓	аналіз комплексу документів, необхідних для проведення сертифікації продукції
✓	прийняття рішення за заявкою із зазначенням схеми сертифікації
✓	укладення договору на проведення робіт із сертифікації
✓	оцінка на відповідність заявленим вимогам відповідно до обраної схеми сертифікації, в т.ч. аудит виробництва, оцінка системи управління, відбір та ідентифікація зразків продукції та їх випробування
✓	визнання сертифіката, виданого іноземними органами сертифікації
✓	аналіз отриманих результатів і ухвалення рішення про можливість видачі сертифіката відповідності
✓	оформлення сертифіката відповідності, укладання сертифікаційного договору на право його застосування та маркування знаком відповідності
✓	інспектування сертифікованої продукції та її виробництва та / або системи управління

Далі



ЗНАКИ ВІДПОВІДНОСТІ



ЛЕКЦІЯ 4

ПРАВОВЕ РЕГУЛЮВАННЯ У ГАЛУЗІ КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ

Мета – знати суб'єкти і об'єкти інформаційних відносин, види інформації, особливості правових відносин в інформаційно-телекомунікаційних системах; вміти встановлювати права та обов'язки розробника та користувача програмного засобу за видом ліцензії.

ПЛАН

1. Законодавство у галузі комп'ютерних технологій
2. Суб'єкти і об'єкти інформаційних відносин
3. Види інформації
4. Комп'ютерна програма як об'єкт авторського права
5. Види ліцензій на програмний засіб

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Науково-технічна інформація
2. Відповідальність за порушення законодавства про інформацію

ЛІТЕРАТУРА

1. Закон України від 02.10.1992 № 2657-ХІІ «Про інформацію»
2. Закон України від 05.07.1994 № 80/94-ВР «Про захист інформації в інформаційно-телекомунікаційних системах»

КОНТРОЛЬНІ ПИТАННЯ

1. Що є об'єктом інформаційних відносин?
2. Які є види інформації з обмеженим доступом?
3. Коли комп'ютерна програма стала об'єктом авторського права в Україні?
4. Які є види ліцензії на програмний засіб?
5. Які є види інформації за змістом?

ОПОРНИЙ КОНСПЕКТ

Про захист інформації в інформаційно-
телекомунікаційних системах № 80/94-ВР
від 05.07.1994

Про авторське право і суміжні
права № 3792-XII від 23.12.1993

Про інформацію № 2657-XII від
02.10.1992



Принципи інформаційних відносин

Напрямки державної інформаційної політики

Суб'єкт

Об'єкт

Право на інформацію

- Принципи інформаційних відносин
- Напрямки державної інформаційної політики
- Суб'єкти і об'єкти інформаційних відносин
- Право на інформацію

Інформація про фізичну особу

Персональні дані - відомості чи їх сукупність про фізичну особу, яка ідентифікована або може бути конкретно ідентифікована. Не допускаються збирання, зберігання, використання та поширення конфіденційної інформації про особу без її згоди, крім випадків, визначених законом, і лише в інтересах національної безпеки, економічного добробуту та прав людини. До конфіденційної інформації про фізичну особу належать, зокрема, дані про її національність, освіту, сімейний стан, релігійні переконання, стан

Інформація довідково-енциклопедичного характеру

Інформація про стан навколишнього середовища

Інформація про товар (роботу, послугу)

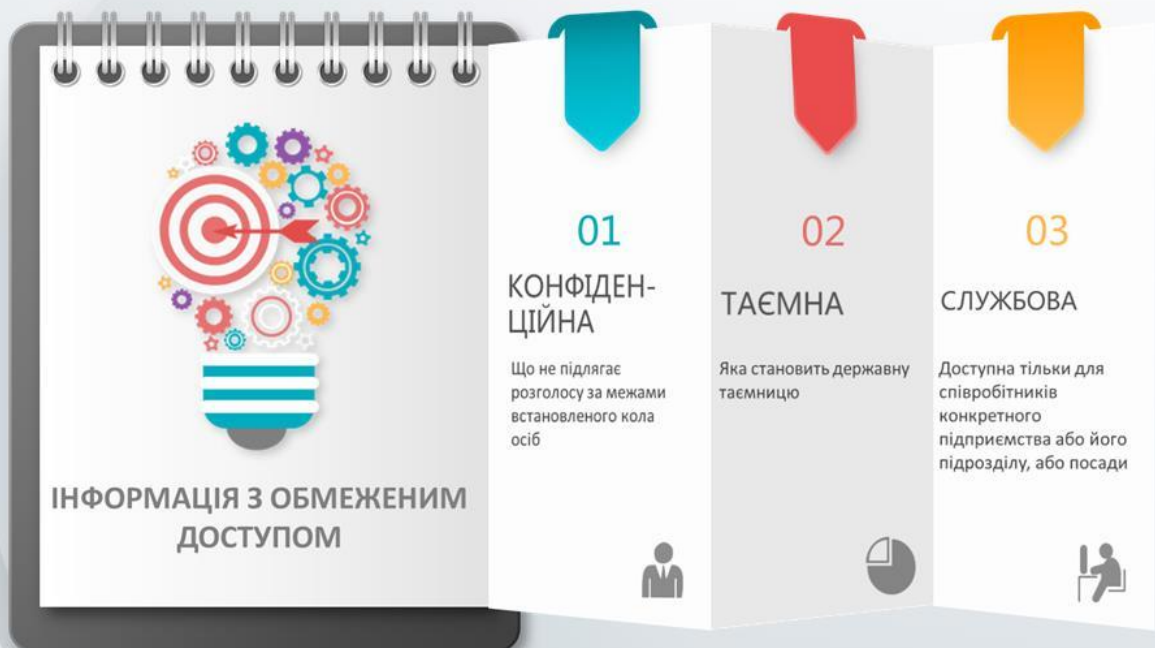
Науково-технічна інформація

Податкова інформація

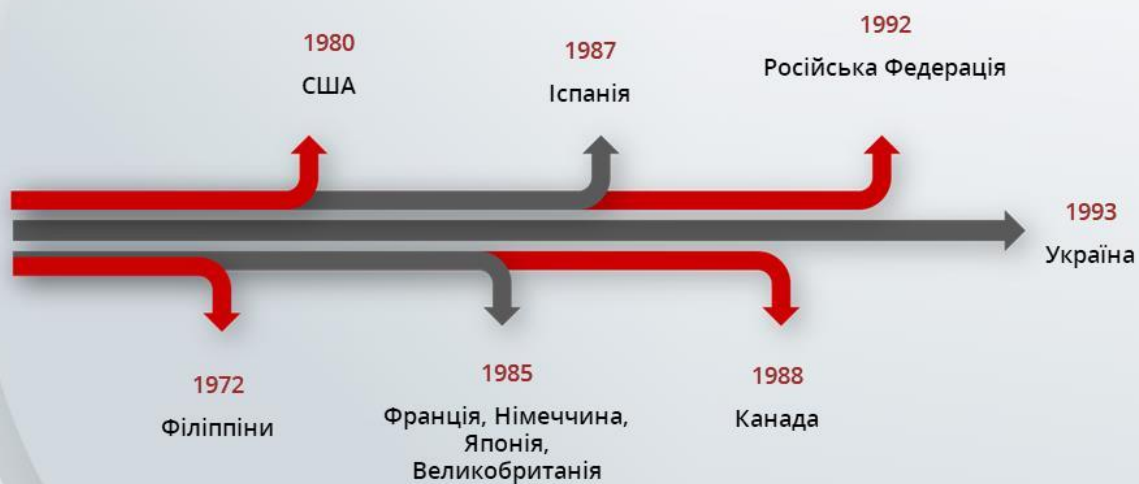
Правова інформація

Статистична інформація

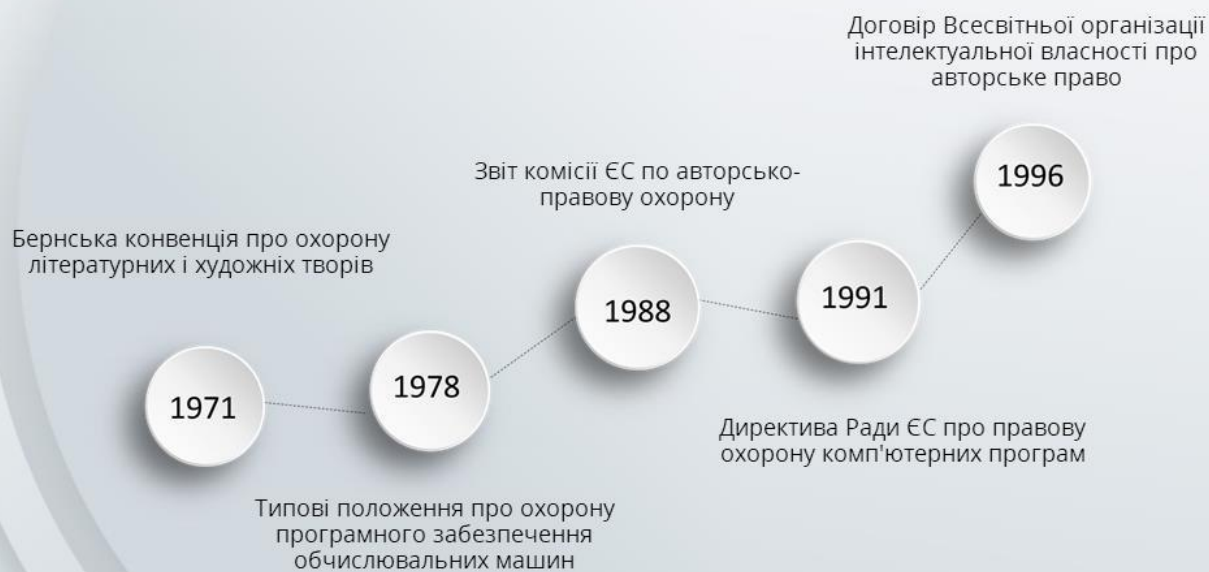
Соціологічна інформація



ВНЕСЕННЯ КОМП'ЮТЕРНИХ ПРОГРАМ ДО ПЕРЕЛІКУ ОБ'ЄКТІВ АВТОРСЬКОГО ПРАВА



ВНЕСЕННЯ КОМП'ЮТЕРНИХ ПРОГРАМ ДО ПЕРЕЛІКУ ОБ'ЄКТІВ АВТОРСЬКОГО ПРАВА





Freeware - безкоштовні програми. Програми без обмеження на (некомерційне) використання. Охороняються авторським правом. Однак, відсутність ціни не означає, що виробник дозволяє її вільно поширювати, він може це і забороняти. Іноді якась програма безкоштовна тільки для домашнього некомерційного використання.

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25



ЛЕКЦІЯ 5 ОСНОВНІ ПОНЯТТЯ ТЕОРІЇ НАДІЙНОСТІ ПРОГРАМНИХ ЗАСОБІВ. МОДЕЛІ НАДІЙНОСТІ

Мета – знати терміни та визначення, критерії та кількісні характеристики надійності програмного забезпечення, способи її забезпечення та підвищення, класифікацію моделей його надійності; вміти визначати категорії тяжкості помилок, встановлювати причини виникнення помилок за їх ознаками, розраховувати надійність програмного забезпечення за різними моделями.

ПЛАН

1. Критерії та кількісні характеристики надійності програмного забезпечення
2. Причини порушення нормального функціонування програми
3. Ознаки виникнення та категорії тяжкості помилок
4. Способи забезпечення та підвищення надійності програмного забезпечення помилок
5. Класифікація моделей надійності програмного забезпечення

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Модель Желінського-Моранди
2. Модель Пуассонівського типу
3. Модель Марківського типу

ЛІТЕРАТУРА

1. Липаев В. В. Сертификация программных средств: Учебник / В. В. Липаев. – М.: Синтег, 2009. – 336 с.

2. Глухова Л. А. Конспект лекцій по курсу «Стандартизація і сертифікація програмного забезпечення» / Л. А. Глухова. – Минск: БГУИР, 2004. – 80 с.

КОНТРОЛЬНІ ПИТАННЯ

1. Які є кількісні характеристики надійності програмного засобу?
2. Якими можуть бути причини порушення нормального функціонування програми?
3. За якими ознаками визначають наявність помилок?
4. Як можна підвищити надійність програми?
5. За якими ознаками класифікують моделі надійності програмних засобів?

ОПОРНИЙ КОНСПЕКТ



ПРИЧИНИ ПОРУШЕННЯ НОРМАЛЬНОГО ФУНКЦІОНУВАННЯ ПРОГРАМИ

помилки, приховані в самій програмі

спотворення вхідної інформації, що підлягає обробці

невірні дії користувача

несправність апаратури, на якій реалізується обчислювальний процес

✓	Помилки обчислень
✓	Логічні помилки
✓	Помилки введення-виведення
✓	Помилки маніпулювання даними
✓	Помилки сумісності з операційною системою
✓	Помилки сполучень з іншими програмами, системними програмами, пристроями комп'ютера, вхідними даними

Далі

✓	спотворення даних на первинних носіях інформації
✓	збої і відмови в апаратурі введення даних з первинних носіїв інформації
✓	шуми і збої в каналах зв'язку при передачі повідомлень по лініях зв'язку



1	передчасне закінчення виконання програми
2	неприпустиме збільшення часу виконання деякої послідовності команд
3	повна втрата або значне спотворення накопичених даних
4	порушення послідовності виклику окремих команд, в результаті чого відбувається пропуск необхідних команд
5	спотворення окремих елементів даних (вхідних, вихідних, проміжних) в результаті обробки спотвореної вихідної інформації

КАТЕГОРІЇ ТЯЖКОСТІ ПОМИЛКИ В ПРОГРАМНОМУ ЗАБЕЗПЕЧЕННІ, ПОРУШЕННЯ ПРАЦЕЗДАТНОСТІ ЯКОГО МОЖУТЬ ПРИЗВЕСТИ ДО КАТАСТРОФІЧНИХ НАСЛІДКІВ

Категорія	Найменування	Наслідки прояви помилки
IV	Катастрофічна	прояв помилки з високою ймовірністю тягне за собою припинення функціонування програмного забезпечення (його відмову) і може викликати пошкодження системи і навколишнього середовища і / або загибель і важкі травми людей
III	Критична	прояв помилки з високою ймовірністю тягне за собою припинення функціонування програмного забезпечення (його відмову), може викликати пошкодження системи і навколишнього середовища, але не загрожує життю і здоров'ю людей
II	Суттєва	прояв помилки тягне за собою зниження ефективності функціонування програмного забезпечення і може викликати припинення функціонування програмного забезпечення (його відмову) без помітного ушкодження системи і загрози життю і здоров'ю людей
I	Несуттєва	прояв помилки може спричинити зниження ефективності функціонування програмного забезпечення і практично не призводить до виникнення відмови в ньому (ймовірність виникнення відмови дуже низька)

КАТЕГОРІЇ ТЯЖКОСТІ ПОМИЛКИ В ПРОГРАМНОМУ ЗАБЕЗПЕЧЕННІ, ПОРУШЕННЯ ПРАЦЕЗДАТНОСТІ ЯКОГО НЕ ПРИЗВОДИТЬ ДО КАТАСТРОФІЧНИХ НАСЛІДКІВ

Категорія	Найменування	Наслідки прояви помилки
III	Критична	прояв помилки з високою ймовірністю тягне за собою припинення функціонування програмного забезпечення (його відмову)
II	Суттєва	прояв помилки тягне за собою зниження ефективності функціонування програмного забезпечення і може викликати припинення функціонування програмного забезпечення (його відмову)
I	Несуттєва	прояв помилки може спричинити зниження ефективності функціонування програмного забезпечення і практично не призводить до виникнення відмови в ньому (ймовірність виникнення відмови дуже низька)

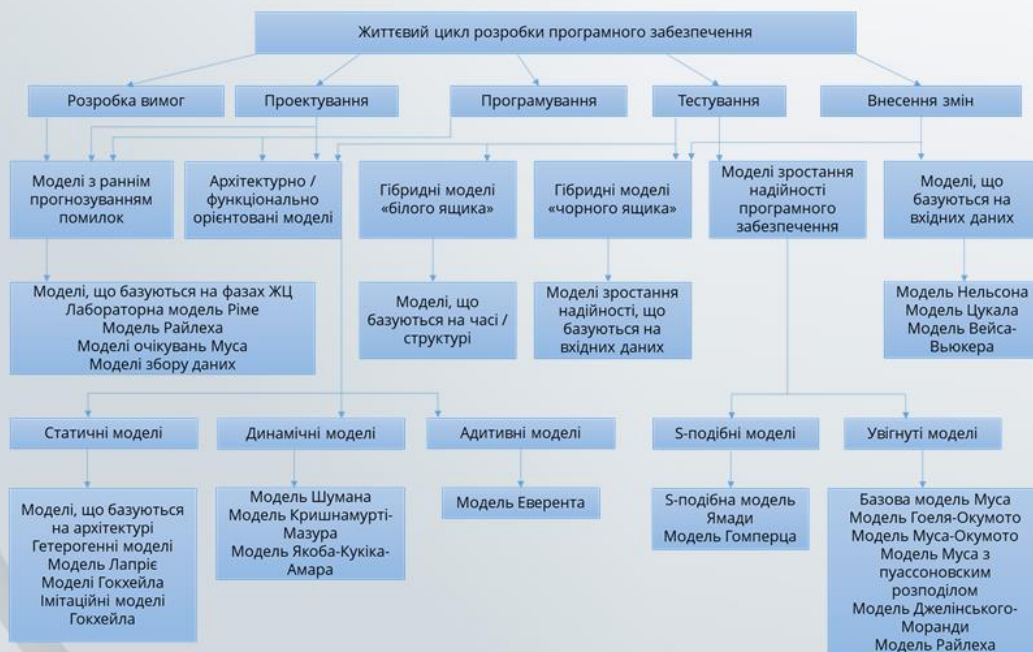
СПОСОБИ ПІДВИЩЕННЯ І ЗАБЕЗПЕЧЕННЯ НАДІЙНОСТІ ПРОГРАМ



КЛАСИФІКАЦІЯ МОДЕЛЕЙ НАДІЙНОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЗА МЕТОДАМИ ОТРИМАННЯ ІНФОРМАЦІЇ



КЛАСИФІКАЦІЯ МОДЕЛЕЙ НАДІЙНОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ЗА СТАДІЯМИ ЖИТТЕВОГО ЦИКЛУ



СТАТИСТИЧНА МОДЕЛЬ МІЛЛСА

$$N = \frac{S \cdot n}{v}$$

1

$$C = \begin{cases} 1, n > k \\ \frac{S}{S+k+1}, n \leq k \end{cases}$$

2

$$C = \begin{cases} 1, n > k \\ \frac{C_S^{v-1}}{C_{S+k+1}^{k+v}}, n \leq k \end{cases}$$

3

$$A_c^b = \frac{c!}{b!(c-b)!}$$

4

Статистична модель Міллса дозволяє оцінити кількість помилок до початку тестування, а також ступінь налаштування програмного забезпечення. Для її застосування до початку тестування в програмне забезпечення навмисно вносять помилки. Потім вважають, що виявлення навмисно внесених і так званих власних помилок програмного забезпечення рівноймовірні.

Для оцінки кількості помилок в програмному забезпеченні до початку тестування використовують вираз (1), де

S - кількість навмисно внесених в програмне забезпечення помилок до початку тестування;

n - кількість помилок програмного забезпечення, виявлених під час тестування;

v - кількість навмисно внесених в програмне забезпечення помилок, виявлених при тестуванні.

Якщо продовжувати тестування, поки всі навмисно внесені помилки будуть виявлені, ступінь налаштування програмного забезпечення можна оцінити за допомогою виразу (2), де де k - максимальна кількість помилок в програмному забезпеченні.

ЕВРИСТИЧНА (ПРОСТА ІНТУЇТИВНА) МОДЕЛЬ

$$N = \frac{n_1 \cdot n_2}{n_{12}}$$

1

$$p_1 = E_1 = \frac{n_1}{N}$$

2

$$p_2 = E_2 = \frac{n_2}{N}$$

3

$$p_{12} = \frac{n_{12}}{N}$$

4

$$\frac{n_{12}}{N} = \frac{n_1}{N} \cdot \frac{n_2}{N}$$

5

Використання евристичної моделі передбачає проведення тестування двома групами фахівців незалежно один від одного, які використовують незалежні тестові набори. Якщо в програмі є N помилок до початку тестування, то вони будуть рівні (1), де n1 і n2 - кількість помилок, які виявили перша і друга група тестувальників відповідно, а n12 - кількість виявлених двома групами тестувальників спільних помилок.

Ефективність тестування кожної з груп і ймовірність виявлення ними помилки дорівнюють (2) і (3). Тоді ймовірність того, що помилка буде виявлена двома групами тестувальників дорівнює (4).

З іншого боку, так як групи діють окремо один від одного, то p12 = p1 * p2. Тоді (5).

МОДЕЛЬ КОРКОЕНА

ОСОБЛИВОСТІ



03 модель містить змінну ймовірність відмов для різних джерел помилок і різну ймовірність їх виправлення



02 в моделі використовують параметри: результат тільки N випробувань, в яких виявлені помилки i-го типу



01 виявлення в ході N випробувань помилки i-го типу відбувається з ймовірністю a_i

$$R = \frac{N_0}{N} + \sum_{i=1}^k \frac{Y_i(N_i - 1)}{N} \quad 1$$

Показник рівня надійності R розраховують за формулою (1), де:

N_0 - число безвідмовних (або безуспішних) випробувань, виконаних в серії з N випробувань

k - відоме число типів помилок

Y_i - ймовірність появи помилок, $Y_i = a_i$ при $N > 0$, $Y_i = 0$ при $N = 0$

МОДЕЛЬ ШУМАНА

ДОПУЩЕННЯ

✓	загальне число команд в програмному забезпеченні на машинній мові постійне
✓	на початку компонувальних випробувань число помилок дорівнює деякій постійній величині, в міру виправлення помилок їх стає менше
✓	під час випробувань нові помилки не вносять
✓	спочатку помилки помітні, за сумарним числом виправлених помилок можна судити про останні
✓	інтенсивність відмов програмного забезпечення пропорційна числу залишкових помилок

Далі

$$\varepsilon_2(\tau) = \frac{M}{I} - \varepsilon_1(\tau) \quad 1$$

$$\varepsilon_1(\tau) = \frac{x}{I} \quad 2$$

$$Z(\tau) = C \cdot \varepsilon_2(\tau) \quad 3$$

$$P(t, \tau) = \exp(-C \cdot \varepsilon_2(\tau) \cdot t) \quad 4$$

$$t = \frac{1}{Z(\tau)} \quad 5$$

ЛЕКЦІЯ 6

ВИМОГИ ДО ЯКОСТІ СИСТЕМ І ПРОГРАМНИХ ЗАСОБІВ ТА ЇЇ ОЦІНЮВАННЯ

Мета – знати еталонні моделі оцінювання якості програмного забезпечення, процес її оцінювання, рівні та методи оцінювання якості програмного забезпечення.

ПЛАН

1. Еталонні моделі оцінювання якості програмного забезпечення
2. Процес оцінювання якості програмного забезпечення
3. Рівні та методи оцінювання якості програмного забезпечення

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Настанова щодо оцінювання для розробників, замовників та незалежних експертів
2. Модуль оцінювання відновлюваності

ЛІТЕРАТУРА

1. ISO/IEC 25000:2014 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE
2. ISO/IEC 25001:2014 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Planning and management
3. ISO/IEC 25020:2019 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Quality measurement framework
4. ISO/IEC 25021:2012 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Quality measure elements
5. ISO/IEC 25022:2016 Systems and software engineering — Systems and software quality requirements and evaluation (SQuaRE) — Measurement of quality in use
6. ISO/IEC 25023:2016 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Measurement of system and software product quality
7. ISO/IEC 25024:2015 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Measurement of data quality
8. ISO/IEC 25030:2019 Systems and software engineering — Systems and software quality requirements and evaluation (SQuaRE) — Quality requirements framework
9. ISO/IEC 25040:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation process
10. ISO/IEC 25041:2012 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation guide for developers, acquirers and independent evaluators
11. ISO/IEC 25040:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation process
12. ISO/IEC 25045:2010 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation module for recoverability
13. ISO/IEC 25051:2014 Software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Requirements for quality of Ready to Use Software Product (RUSP) and instructions for testing
14. ISO/IEC TR 25060:2010 Systems and software engineering — Systems and software product Quality Requirements and Evaluation (SQuaRE) — Common Industry Format (CIF) for usability: General framework for usability-related information
15. ISO/IEC 25062:2006 Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Common Industry Format (CIF) for usability test reports

16. ISO/IEC/IEEE 90003:2018 Software engineering — Guidelines for the application of ISO 9001:2015 to computer software

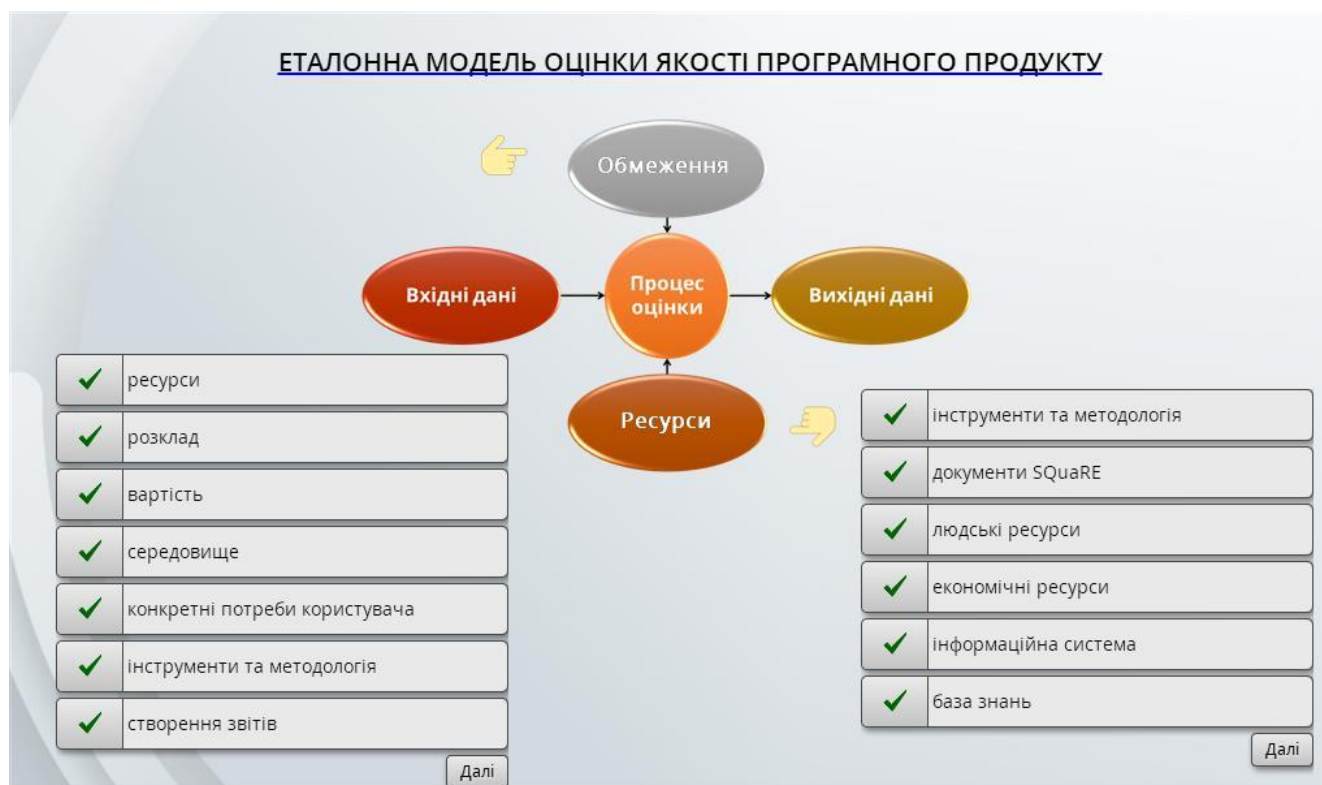
17. ISO/IEC 29155-1:2017 Systems and software engineering — Information technology project performance benchmarking framework — Part 1: Concepts and definitions

18. ISO/IEC 29155-2:2013 Systems and software engineering — Information technology project performance benchmarking framework — Part 2: Requirements for benchmarking

КОНТРОЛЬНІ ПИТАННЯ

1. З яких компонентів складається модель оцінки якості програмного засобу?
2. Які етапи процесу оцінки якості програмного засобу?
3. Яка вхідна інформація потрібна для кожного етапу?
4. Яка вихідна інформація потрібна для кожного етапу?
5. Які методи використовують для оцінки якості програмного засобу?

ОПОРНИЙ КОНСПЕКТ



Процес оцінки якості програмного продукту



Встановити мету оцінки

Мета оцінки якості програмного продукту повинна бути задокументована як підстава для подальших дій і завдань оцінки. Кінцева мета полягає в тому, щоб переконатися, що продукт забезпечує необхідну якість, що він відповідає встановленим і неявним потребам користувачів (включаючи операторів, споживачів результатів програмного забезпечення і фахівців з обслуговування програмного забезпечення) на всіх етапах життєвого циклу. Якість програмного продукту може бути оцінена або як якість проміжного продукту, або як якість кінцевого продукту. Мета оцінки якості проміжного програмного продукту:

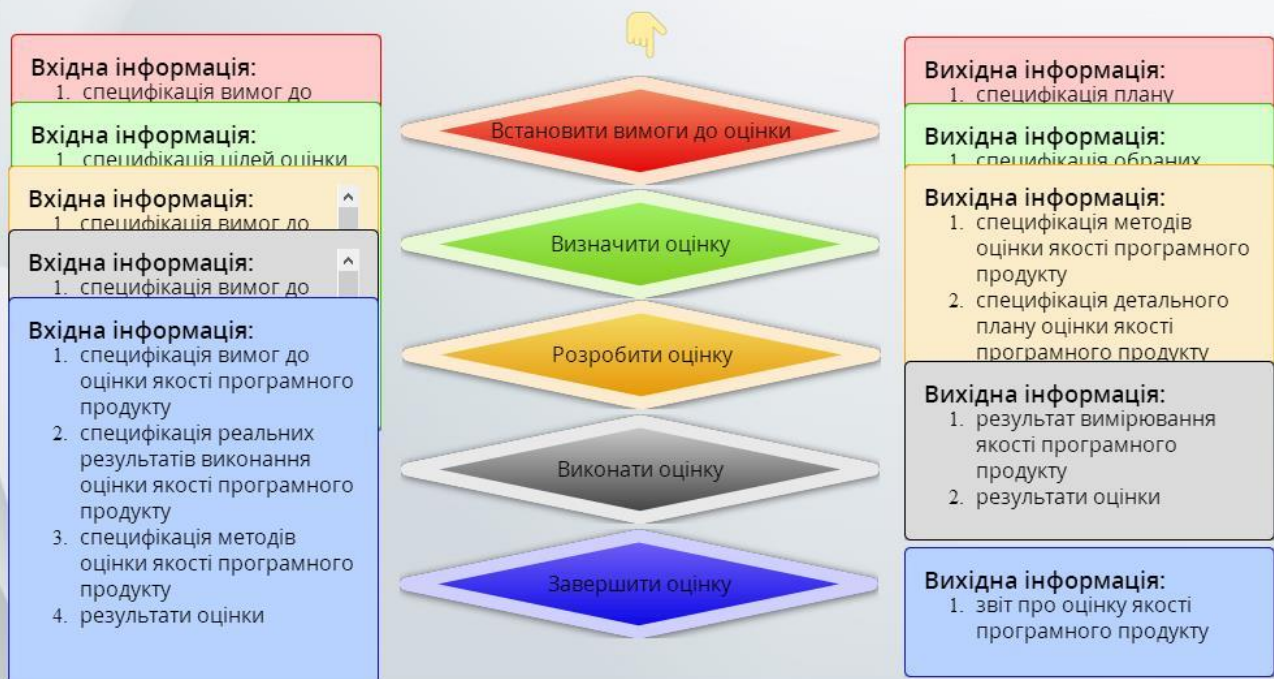
- забезпечити якість продукту;
- прийняти рішення про приймання проміжного програмного продукту від субпідрядника;
- оцінити поточну доцільність проекту розробки;
- прийняти рішення про завершення етапу життєвого циклу і часу переходу продукту до наступного етапу;
- передбачити або оцінити якість кінцевого програмного продукту;
- зібрати інформацію про проміжні програмні продукти, для того щоб контролювати і управляти процесом.

Мета оцінки якості кінцевого програмного продукту може полягати в наступному:

- прийняти рішення про приймання продукту;
- визначити термін виходу продукту;
- порівняти продукт з конкурентоспроможними продуктами;



ВХІДНА І ВИХІДНА ІНФОРМАЦІЯ ПРИ ОЦІНЦІ ЯКОСТІ ПРОГРАМНОГО ПРОДУКТУ



ПРИКЛАДИ МЕТОДІВ ОЦІНКИ ЯКОСТІ ПРОГРАМНОГО ПРОДУКТУ



ЛЕКЦІЯ 7 МОДЕЛІ ЯКОСТІ

Мета – знати: структуру моделі якості, моделі зовнішньої та внутрішньої якості, моделі якості під час використання.

ПЛАН

1. Структура моделі якості
2. Моделі зовнішньої та внутрішньої якості
3. Модель якості під час використання

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Моделі якості системи та програмних засобів
2. Модель якості даних

ЛІТЕРАТУРА

1. ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models
2. ISO/IEC 25012:2008 Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Data quality model

КОНТРОЛЬНІ ПИТАННЯ

1. Які цілі моделей якості програмного продукту?
2. З яких показників складається модель якості під час використання?
3. З яких показників складається модель якості програмного продукту?

4. Які цільові об'єкти моделі якості?

5. З яких показників складається модель життєвого циклу якості системи?

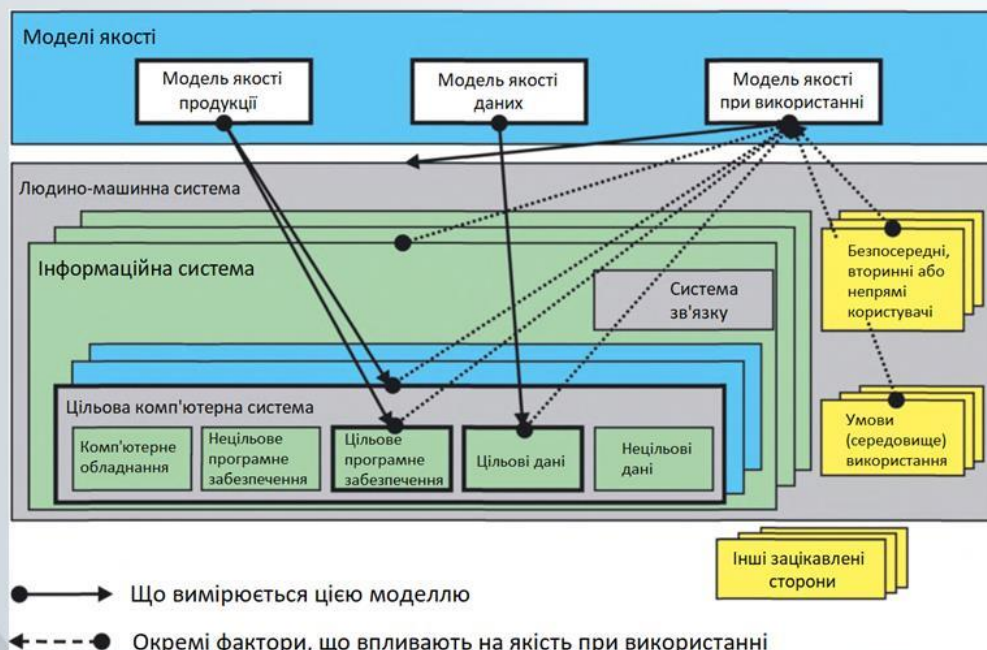
ОПОРНИЙ КОНСПЕКТ

СТРУКТУРА, ЯКА ВИКОРИСТОВУЄТЬСЯ ДЛЯ МОДЕЛЕЙ ЯКОСТІ ПРОГРАМНОГО ПРОДУКТУ



i

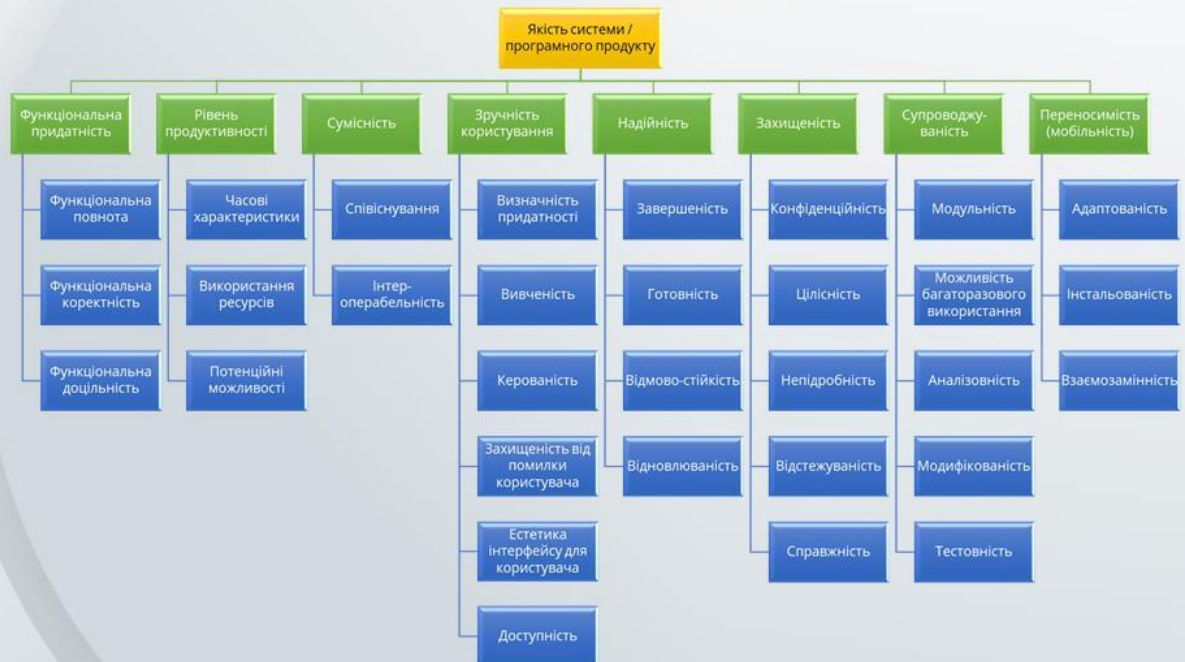
ЦІЛІ МОДЕЛЕЙ ЯКОСТІ



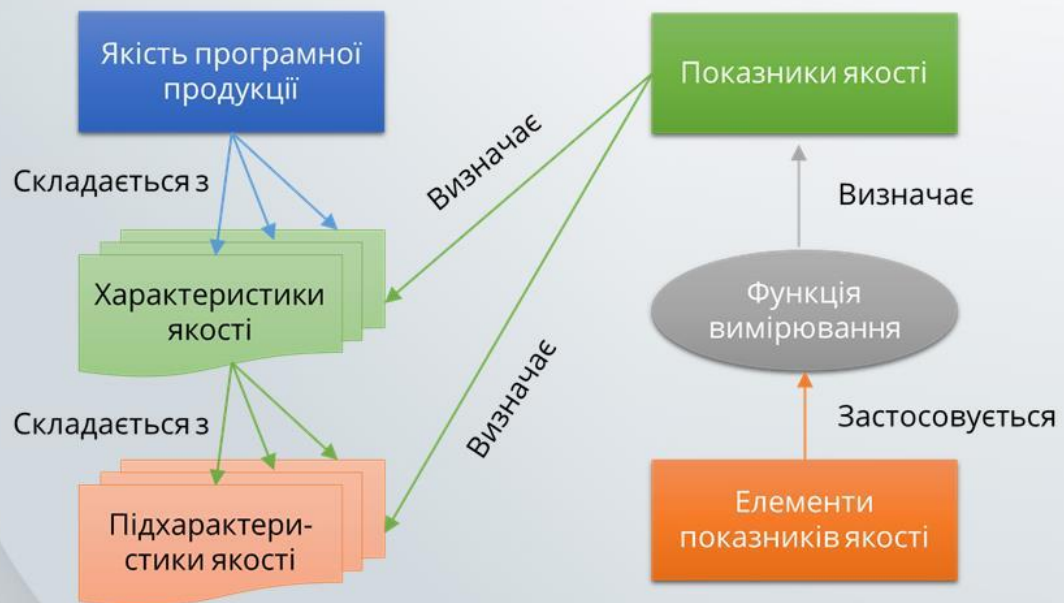
МОДЕЛЬ ЯКОСТІ ПІД ЧАС ВИКОРИСТАННЯ



МОДЕЛЬ ЯКОСТІ ПРОДУКТУ



МОДЕЛЬ ВИМІРЮВАННЯ ЯКОСТІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

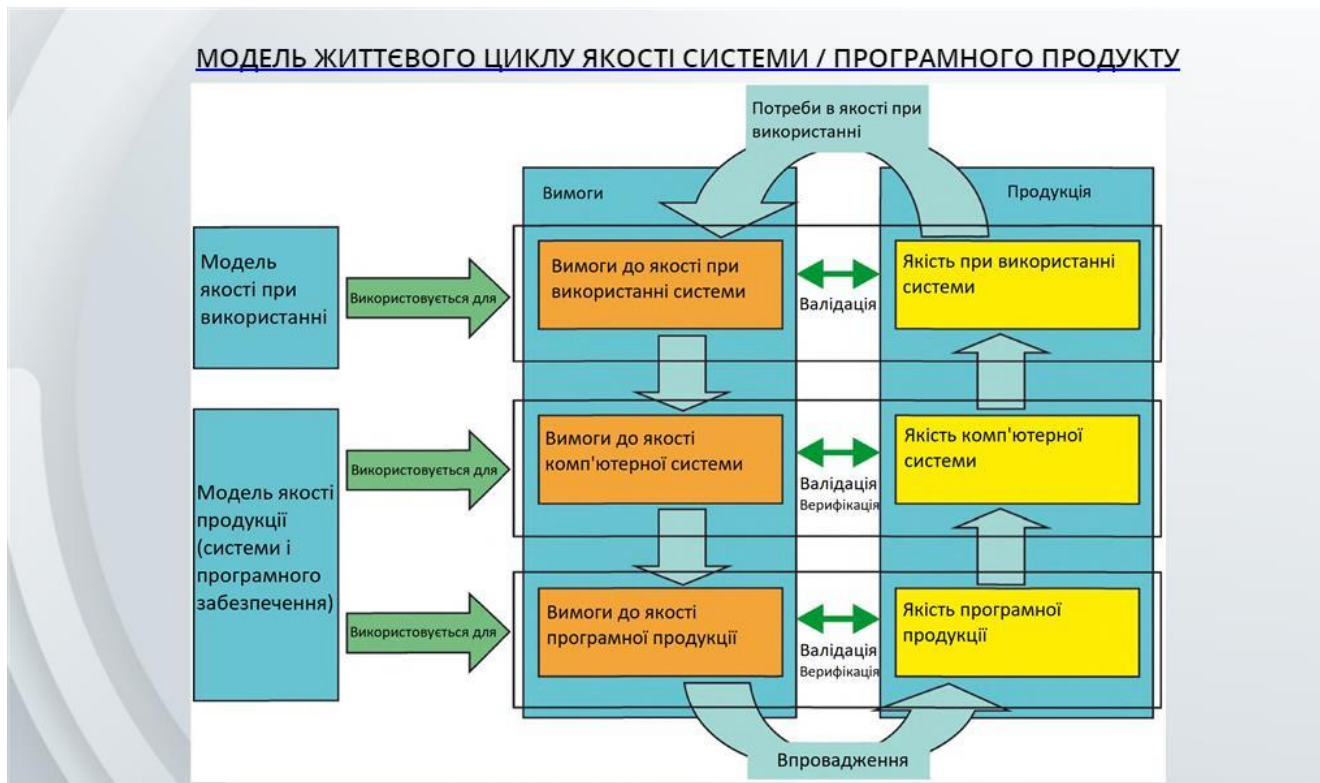


ЦІЛЬОВІ ОБ'ЄКТИ МОДЕЛІ ЯКОСТІ ТА ЇХ ВЗАЄМОЗВ'ЯЗОК



i

МОДЕЛЬ ЖИТТЄВОГО ЦИКЛУ ЯКОСТІ СИСТЕМИ / ПРОГРАМНОГО ПРОДУКТУ



ЛЕКЦІЯ 8 МЕТРИКИ ЯКОСТІ

Мета – знати типи метрик якості програмного забезпечення, типи мір для вимірювання його якості, методологію метричного аналізу якості програмного забезпечення.

ПЛАН

1. Типи метрик якості програмного забезпечення
2. Типи мір для вимірювання якості програмного забезпечення
3. Види шкал
4. Методологія метричного аналізу якості програмного забезпечення

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Зовнішні метрики
2. Внутрішні метрики
3. Метрики якості під час використання

ЛІТЕРАТУРА

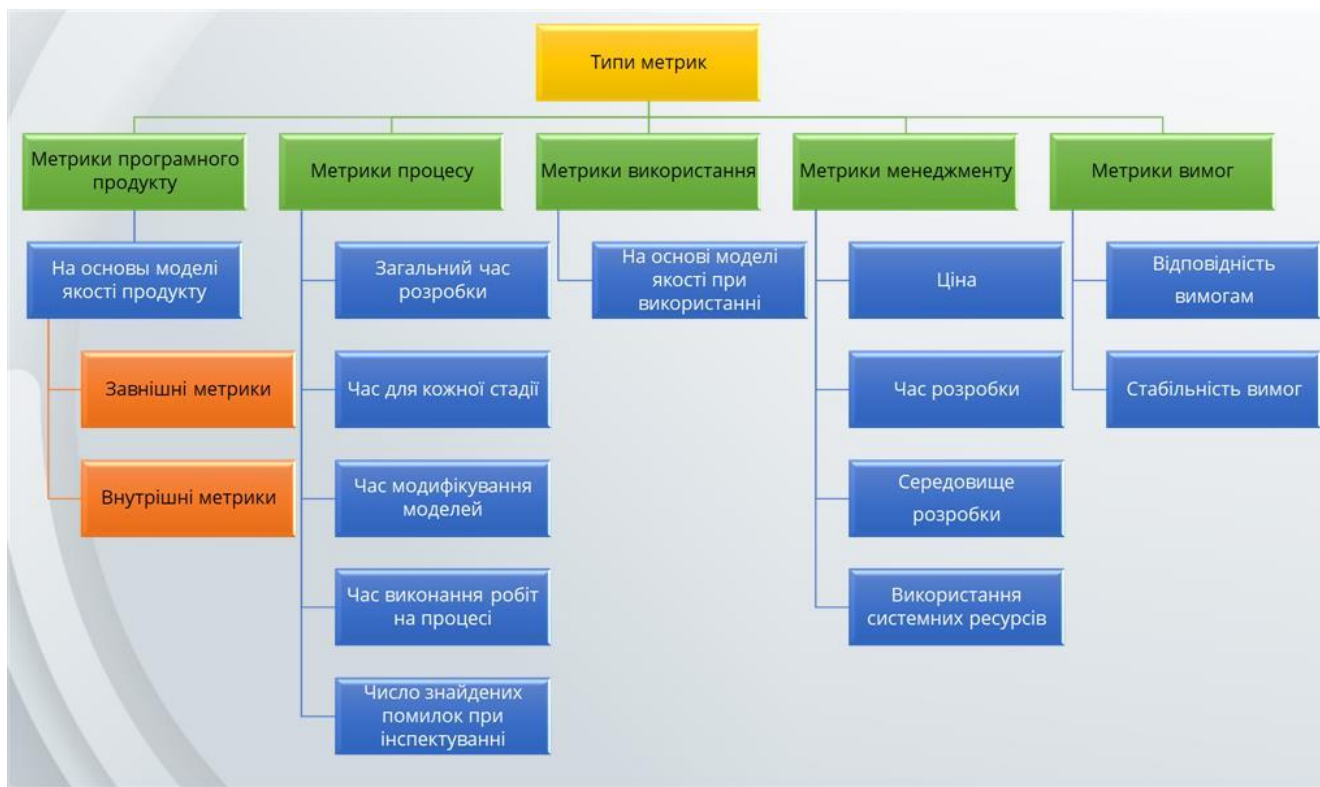
1. ISO/IEC 25020:2019 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Quality measurement framework
2. ISO/IEC 25021:2012 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Quality measure elements
3. ISO/IEC 25022:2016 Systems and software engineering — Systems and software quality requirements and evaluation (SQuaRE) — Measurement of quality in use
4. ISO/IEC 25023:2016 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Measurement of system and software product quality

5. ISO/IEC 25024:2015 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Measurement of data quality
6. ISO/IEC 25030:2019 Systems and software engineering — Systems and software quality requirements and evaluation (SQuaRE) — Quality requirements framework
7. ISO/IEC 25040:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation process
8. ISO/IEC 25041:2012 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation guide for developers, acquirers and independent evaluators
9. ISO/IEC 25040:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation process
10. ISO/IEC 25045:2010 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Evaluation module for recoverability
11. ISO/IEC 25051:2014 Software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Requirements for quality of Ready to Use Software Product (RUSP) and instructions for testing
12. ISO/IEC TR 25060:2010 Systems and software engineering — Systems and software product Quality Requirements and Evaluation (SQuaRE) — Common Industry Format (CIF) for usability: General framework for usability-related information
13. ISO/IEC 25062:2006 Software engineering — Software product Quality Requirements and Evaluation (SQuaRE) — Common Industry Format (CIF) for usability test reports
14. ISO/IEC/IEEE 90003:2018 Software engineering — Guidelines for the application of ISO 9001:2015 to computer software

КОНТРОЛЬНІ ПИТАННЯ

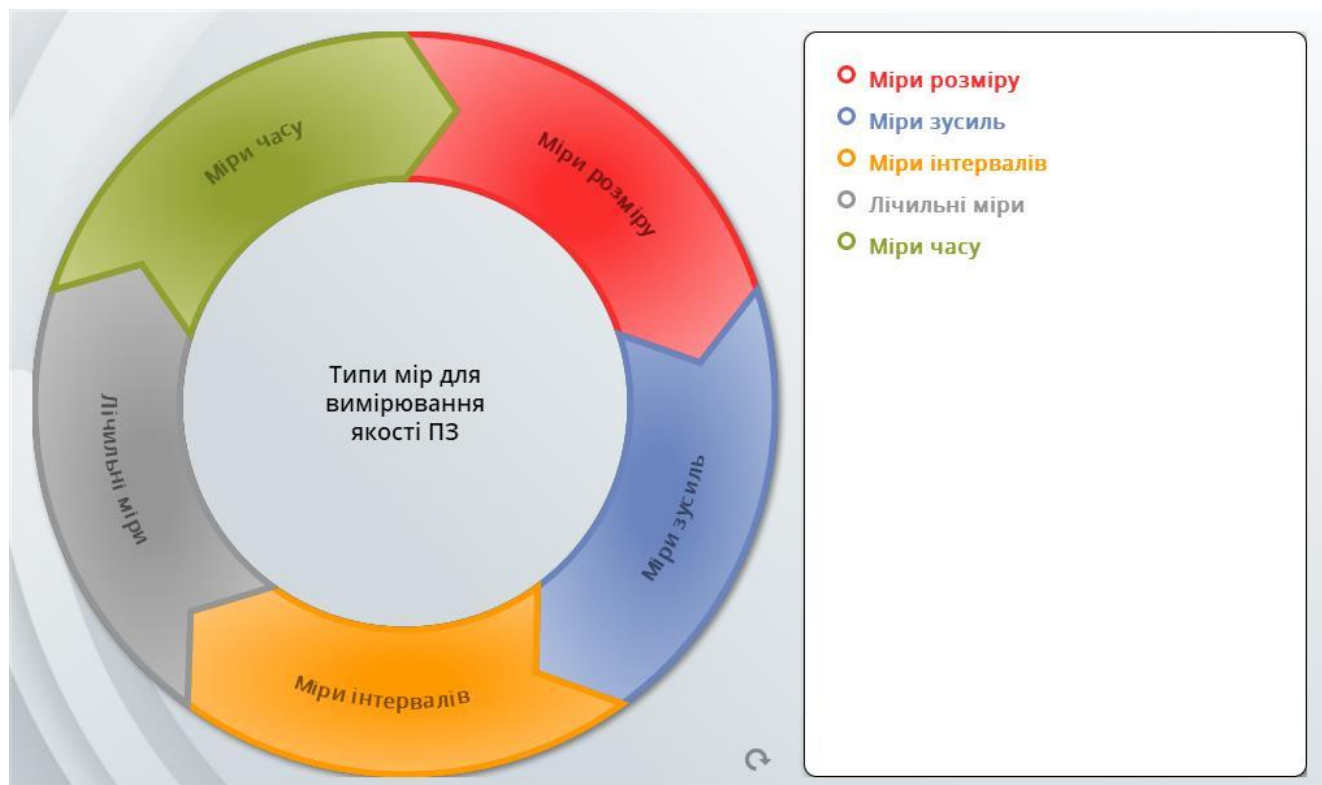
1. Які є типи метрик?
2. Якими методами визначають показники якості програмного продукту?
3. Які існують типи мір для вимірювання якості програмного забезпечення?
4. Які існують типи шкал?
5. Яку інформацію наводять у метриці якості?

ОПОРНИЙ КОНСПЕКТ



МЕТОДИ ВИЗНАЧЕННЯ ПОКАЗНИКІВ ЯКОСТІ ПРОГРАМНОГО ПРОДУКТУ





ТИПИ МІР ДЛЯ ВИМІРЮВАННЯ ЯКОСТІ ПРОГРАМНОГО ПРОДУКТУ

Стиль	Опис	Приклад
Стиль нормалізації за розміром		
(Час) / (Розмір)	для подання часової ефективності або виробничих зусиль на одиницю об'єму	час виконання на інструкцію вихідного коду, час виправлення на інструкцію вихідного коду
(Кількість) / (Розмір)	для подання щільності	щільність помилок, тестове покриття
(Розмір) / (Розмір)	для представлення відношення розміру певного фрагмента до розміру цілого	цикломатичне число модуля, процентиля обсягу модулів, що містять помилки, до загального обсягу
Стиль нормалізації за часом		
(Розмір) / (Время)	для подання тренда величини в часі	число рядків розробленого коду в місяць
(Кількість) / (Час)	для подання частоти в одиницю часу	число транзакцій в секунду, число змінених інструкцій вихідного коду в місяць
(Час) / (Час)	для представлення відношення часу наявних подій до загального часу спостереження	відношення корисного часу роботи до часу функціонування системи
Стиль нормалізації за кількістю		
(Розмір) / (Кількість)	для подання сфери охоплення (діапазону змін)	середнє покриття тестами виконуваних інструкцій програми
(Час) / (Кількість)	для подання інтервалу часу	середній час між відмовами, середній час виправлення однієї помилки
(Кількість) / (Кількість)	для представлення відношення числа певних подій до їх загальної кількості	число успішних спроб користувача виконати дію до загальної кількості спроб

ПРИКЛАДИ МІР ДЛЯ ПОКАЗНИКІВ ЯКОСТІ ПРОГРАМНОГО ПРОДУКТУ

Характеристики якості		Міра	Шкала
Показник	Надійність		
<i>Атрибут</i>	<i>Завершеність</i>		
Елементи	напрацювання на відмову при відсутності рестарту	години	10-1000
<i>Атрибут</i>	<i>Стійкість</i>		
Елементи	напрацювання на відмову при наявності автоматичного рестарту	години	10-1000
	відносні ресурси на забезпечення надійності та рестарту	%	10-90
<i>Атрибут</i>	<i>Відновлюваність</i>		
Елементи	тривалість відновлення	хвилини	10 ⁻² - 10
<i>Атрибут</i>	<i>Доступність-готовність:</i>		
Елементи	відносний час працездатного функціонування	ймовірність	0,7 - 0,99

ПРИКЛАДИ МІР ДЛЯ ПОКАЗНИКІВ ЯКОСТІ ПРОГРАМНОГО ПРОДУКТУ

Характеристики якості		Міра	Шкала
Показник	Практичність		
<i>Атрибут</i>	<i>Зрозумілість</i>		
Елементи	чіткість концепції	порядкова	Відмінна, хороша, задов., незадов.
	демонстраційні можливості		
	наочність і повнота документації		
<i>Атрибут</i>	<i>Простота використання</i>		
Елементи	простота управління функціями	порядкова	Відмінна, хороша, задов., незадов.
	комфортність експлуатації		
	середній час введення завдання	секунди	1-1000

Види шкал



Номінальна шкала

Шкала порядків (рангів)

Шкала інтервалів (різниць)

Шкала відношень

Абсолютна шкала



Такі шкали вимірювань використовують для класифікації емпіричних об'єктів, властивості яких проявляються лише в відношенні еквівалентності, причому ці властивості не можна вважати фізичними величинами. Номінальні шкали, або, як їх ще називають шкали найменувань, так само називають шкалами вимірювань, або шкалами класифікацій. Це найпростіший тип шкал, заснований на приписуванні якісним властивостям

Інформація для опису елементів показника якості програмного продукту



Ім'я елемента

Елемент показника якості повинен мати унікальне ім'я і при необхідності повинен бути ідентифікований порядковим номером. У більшості випадків, ім'я починається з "число ..." (шкала відносин)".

1

2

3

4

5

6

7

8

9

10

11

12

13

14

ПРИКЛАД МЕТРИКИ

Ім'я метрики	Ефективність
Призначення метрики	Яка частка завдання завершена коректно?
Спосіб застосування	Тестування користувача
Вимірювання, формула, елементи даних для обчислення	$M1 = A \times B$ A = Кількість (повністю) = частка заданих цілей на виході B = Кількість (повністю) = ступінь досягнутих заданих цілей, які були представлені на виході
Інтерпретація вимірюваного значення	$0 \leq M1 \leq 1$ Чим ближче до 1.0, тим краще
Тип шкали метрики	Абсолютна
Тип вимірювання	A = число B = число $M1 = X = \text{число} \times \text{число}$
Вихідні дані для вимірювання	Звіт операції (тесту), запис моніторингу користувача
Процес життєвого циклу ПЗ	Валідація, тестування кваліфікації, експлуатація
Виконавець	Користувач, проектувальник інтерфейсу

i

ПРОЕКТНО-ОРІЄНТОВАНІ МЕТРИКИ ЯКОСТІ: МЕТОДОЛОГІЯ АНАЛІЗУ



ЛЕКЦІЯ 9

ПРОЦЕСИ ЖИТТЄВОГО ЦИКЛУ СИСТЕМ

Мета – знати моделі та етапи життєвого циклу систем.

ПЛАН

1. Моделі та етапи життєвого циклу систем
2. Процеси угоди
3. Процеси проекту
4. Технічні процеси

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Людино-орієнтовні процеси проектування систем
2. Оцінювання програмного процесу

ЛІТЕРАТУРА

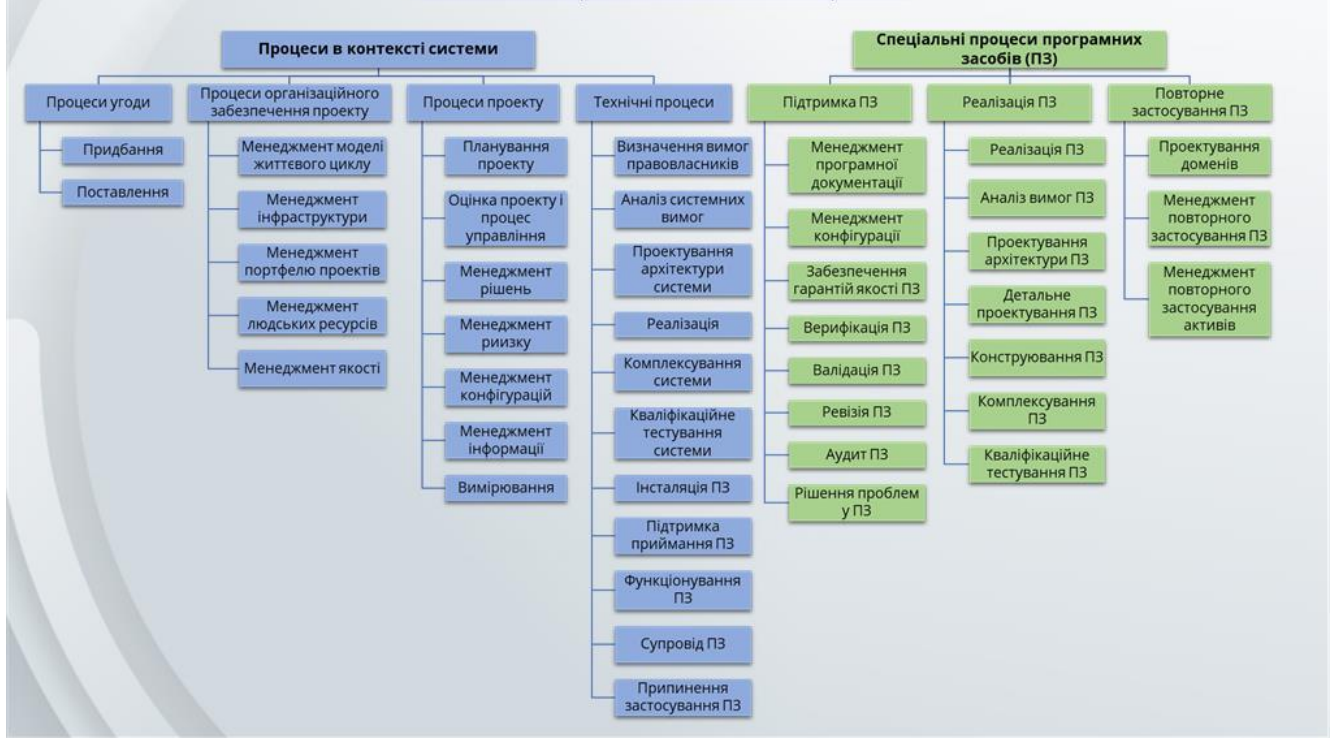
1. ISO/IEC/IEEE 12207:2017 Systems and software engineering — Software life cycle processes
2. ISO/IEC 15288:2014 Systems and software engineering — System life cycle processes
3. ISO/IEC TR 24748-3:2011 Systems and software engineering — Life cycle management — Part 3: Guide to the application of ISO/IEC 12207 (Software life cycle processes)

КОНТРОЛЬНІ ПИТАННЯ

1. З яких процесів складається життєвий цикл системи?
2. З яких етапів складається процес угоди?
3. З яких етапів складається процес організаційного забезпечення проекту?
4. З яких етапів складається процес проекту?
5. З яких етапів складаються технічні процеси?

ОПОРНИЙ КОНСПЕКТ

ГРУПИ ПРОЦЕСІВ ЖИТТЕВОГО ЦИКЛУ



АТРИБУТИ ОПИСУ КОЖНОГО ПРОЦЕСУ



ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ПРОЦЕСИ УГОДИ. ПРИДБАННЯ



- ☒ Мета
- ☐ Вихідні результати
- ☒ Діяльність і завдання

✓	описати потребу
✓	визначити системні вимоги
✓	<u>вибрати варіант придбання</u>
✓	<u>скласти план придбання</u>
✓	визначити умови приймання
✓	<u>документувати вимоги до придбання</u>

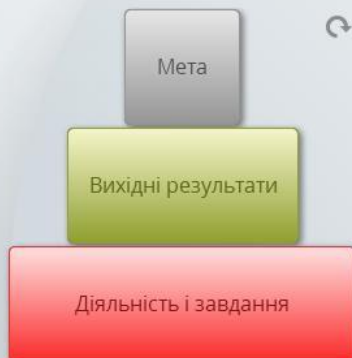
Варіанти придбання:

План придбання:

Документувати вимоги:

- системні вимоги
- формулювання області застосування
- інструкції для претендентів
- перелік програмних продуктів
- терміни і умови
- контроль підрядників
- технічні обмеження

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ПРОЦЕСИ УГОДИ. ПОСТАЧАННЯ



- ☐ Мета
- ☒ Вихідні результати
- ☒ Діяльність і завдання

Виконання контракту

розглянути вимоги замовника

1 2 3 4 5 6 7 8

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ПРОЦЕСИ ОРГАНІЗАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ ПРОЕКТУ.
МЕНЕДЖМЕНТ МОДЕЛІ ЖИТТЕВОГО ЦИКЛУ

Мета

визначення, супровід і забезпечення гарантії наявності політик, процесів і моделей життєвого циклу і процедур для використання організацією

Завдання

запровадження процесів

оцінка процесів

Вдосконалення процесів

Вихідні результати

надаються політики і процедури менеджменту і розгортання моделей і процесів життєвого циклу

визначаються обов'язки, відповідальність і повноваження менеджменту життєвого циклу

визначаються, супроводжуються і удосконалюються процеси, моделі і процедури життєвого циклу для застосування організацією

здійснюється процес удосконалень в порядку встановлених пріоритетів

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ПРОЦЕСИ ОРГАНІЗАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ ПРОЕКТУ

Менеджмент інфраструктури



Мета

постачання проекту забезпечуваною інфраструктурою і послугами для підтримки організації та цілей проекту протягом всього життєвого циклу

Діяльність і завдання

Вихідні результати

Менеджмент якості



Мета

Діяльність і завдання

Вихідні результати

гарантія того, що продукти, послуги та реалізації процесів життєвого циклу відповідають цілям організації в області якості і задовольняють замовника

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ПРОЦЕСИ ОРГАНІЗАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ ПРОЕКТУ. МЕНЕДЖМЕНТ ПОРТФЕЛЯ ПРОЕКТІВ

МЕТА

ініціація і підтримка необхідних, достатніх і відповідних проектів для виконання стратегічних цілей організації

ВИХІДНІ РЕЗУЛЬТАТИ

- 1 уточнюються, розставляються за пріоритетам і вибираються можливості, інвестиції або потреби ділової сфери з урахуванням ризиків
- 2 визначаються і розподіляються ресурси і кошти для кожного проекту
- 3 визначаються повноваження і відповідальність керівництва проекту
- 4 підтримуються проекти, що задовольняють умовам угоди і вимогам правовласників
- 5 переорієнтуються або припиняються проекти, що не відповідають умовам угоди або вимогам правовласників

ДІЯЛЬНІСТЬ І ЗАВДАННЯ

✓	Ініціація проекту
✓	Оцінка проекту
✓	Закриття проекту

Далі

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ПРОЦЕСИ ОРГАНІЗАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ ПРОЕКТУ. МЕНЕДЖМЕНТ ЛЮДСЬКИХ РЕСУРСІВ



- Мета
- Діяльність та завдання
- Вихідні результати

- ✓ Планувати вимоги для управління активами знань організації: визначення інфраструктури та навчання для підтримки вкладників і користувачів активів знань організації, схеми класифікації активів і критеріїв активів
- ✓ Заснувати мережу експертів в межах організації: ідентифікація експертів організації, список областей їх експертизи та ідентифікація доступної інформації в межах схеми класифікації
- ✓ Встановлювати механізм підтримки обміну інформацією між експертами та проектами організації
- ✓ Виконувати управління конфігурацією активів

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ПРОЦЕСИ ПРОЕКТУ. ПЛАНУВАННЯ ПРОЕКТУ

МЕТА

складання та доведення до зацікавлених сторін ефективного і здійсненого плану

ВИХІДНІ РЕЗУЛЬТАТИ

- step 01** визначається область проведення робіт по проекту
- step 02** оцінюється можливість досягнення кінцевих цілей проекту з наявними ресурсами і обмеженнями
- step 03** визначаються розміри і оцінюються завдання і ресурси, необхідні для виконання роботи
- step 04** ідентифікуються інтерфейси між елементами в проекті і з іншими проектами і підрозділами організації
- step 05** розробляються плани реалізації проекту
- step 06** активізуються плани реалізації проекту

ДІЯЛЬНІСТЬ І ЗАВДАННЯ

✓	Ініціація проекту
✓	<u>Планування проекту</u>
✓	Активізація проекту

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ПРОЦЕСИ ПРОЕКТУ

Оцінка проекту і процес управління



Мета	визначення стану проекту і гарантія того, що проект виконується відповідно до планів і графіків робіт в межах бюджету і задовольняє технічним параметрам	Діяльність і завдання	Вихідні результати
------	--	-----------------------	--------------------



- Мета
- Вихідні результати
- Діяльність і завдання

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ПРОЦЕСИ ПРОЕКТУ

Менеджмент ризику



Мета

постійне визначення, аналіз, обробка та моніторинг ризиків

Діяльність і завдання

Вихідні результати

Менеджмент конфігурації



Мета

Діяльність і завдання

Вихідні результати

встановлення і підтримання цілісності всіх ідентифікованих вихідних результатів проекту або процесу забезпечення доступу до них будь-якої зацікавленої сторони

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ПРОЦЕСИ ПРОЕКТУ. МЕНЕДЖМЕНТ ІНФОРМАЦІЇ

МЕТА

своєчасне надання зацікавленим сторонам релевантної, повної, достовірної та конфіденційної інформації протягом та після завершення життєвого циклу системи



- 01 визначається інформація, що підлягає управлінню
- 02 визначаються форми подання інформації
- 03 інформація перетворюється і розподіляється відповідно до вимог
- 04 визначається статус інформації
- 05 визначити інформаційні потреби менеджменту протягом життєвого циклу системи, політикою організації або протягом певного періоду часу
- 06 інформація стає доступною

ДІЯЛЬНІСТЬ І ЗАВДАННЯ

1. Планування менеджменту інформації

2. Визначення інформаційних потреб

3. визначити права, обов'язки і зобов'язання, що стосуються зберігання, передачі і доступу до інформаційних блоків

4. забезпечити надання інформації менеджменту інформації

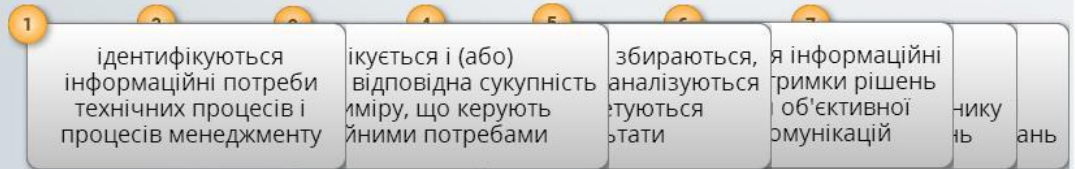
5. передачі і пошуку інформації

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ПРОЦЕСИ ПРОЕКТУ. ВИМІРЮВАННЯ

МЕТА

збір, аналіз і складання звітів про дані, що відносяться до розроблених продуктів і процесів, реалізованих в межах певного організаційного підрозділу, для підтримки ефективного менеджменту процесів і об'єктивної демонстрації якості цих продуктів

ВИХІДНІ РЕЗУЛЬТАТИ



ДІЯЛЬНІСТЬ І ЗАВДАННЯ

✓	Планування вимірювань
✓	Виконання вимірювань
✓	Оцінювання вимірювань

Планування вимірювань:

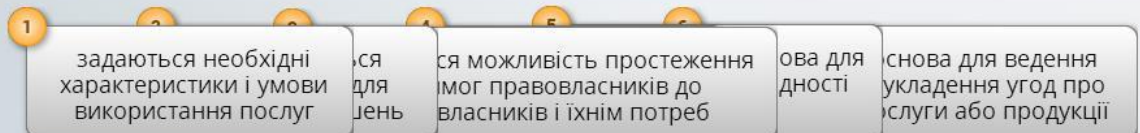
- описати характеристики організації, що проводить вимірювання
- ідентифікувати і розподілити за пріоритетами потреби в інформації
- вибрати і документувати одиниці виміру, що задовольняють інформаційним потребам
- визначати процедури збору даних, аналізу та подання звітів
- визначати критерії для оцінки інформаційних продуктів і

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ТЕХНІЧНІ ПРОЦЕСИ. ВИЗНАЧЕННЯ ВИМОГ ПРАВОВЛАСНИКІВ

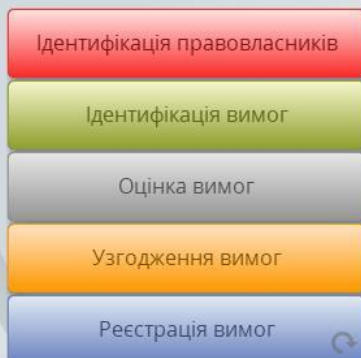
МЕТА

виявити вимоги до системи, виконання яких може забезпечувати надання послуг, необхідних користувачам та іншим правовласникам в заданому середовищі застосування

ВИХІДНІ РЕЗУЛЬТАТИ



ДІЯЛЬНІСТЬ І ЗАВДАННЯ



- Ідентифікація правовласників
- Ідентифікація вимог
- Оцінка вимог
- Узгодження вимог
- Реєстрація вимог

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ТЕХНІЧНІ ПРОЦЕСИ

Аналіз системних вимог



Мета

перетворення певних вимог правовласників в сукупність необхідних системних технічних вимог, якими будуть керуватися в проекті системи

Діяльність і завдання

Вихідні результати

ПРОЕКТУВАННЯ
АРХІТЕКТУРИ СИСТЕМИ

Мета

Створення
архітектури

Оцінювання
архітектури

Вихідні результати

- ☒ Мета
- ☐ Діяльність і завдання
- ☐ Вихідні результати

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ТЕХНІЧНІ ПРОЦЕСИ

Реалізація

Мета - створення заданих елементів системи;
окремий випадок процесу реалізації див. у темі 3.2

КОМПЛУКСУВАННЯ

Вихідні результати

Комплексу-
вання

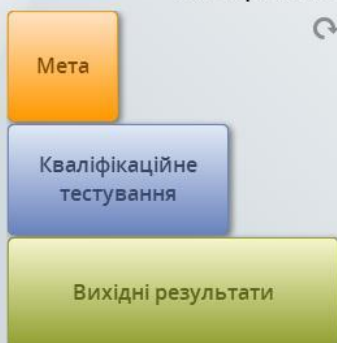
Готовність до
тестування

Мета

- ☐ Вихідні результати
- ☐ Діяльність і завдання
- ☒ Мета

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ТЕХНІЧНІ ПРОЦЕСИ

КВАЛІФІКАЦІЙНЕ
ТЕСТУВАННЯ



- Мета
- Діяльність і завдання
- Вихідні результати



- Мета
- Вихідні результати
- Діяльність і завдання

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ТЕХНІЧНІ ПРОЦЕСИ. ПІДТРИМКА ПРИЙНЯТТЯ ПРОГРАМНИХ ЗАСОБІВ

МЕТА

сприяння стороні, що набуває, в забезпеченні впевненості в тому, що продукт відповідає заданим вимогам

ВИХІДНІ РЕЗУЛЬТАТИ

- 01 продукт комплектується і поставляється покупцю
- 02 підтримуються приймальні випробування і ревізії, що проводяться покупцем
- 03 продукт застосовується за призначенням в середовищі замовника
- 04 проблеми, виявлені протягом приймання, ідентифікуються і передаються відповідальним за їх рішення

ДІЯЛЬНІСТЬ І ЗАВДАННЯ

1.

Підтримка приймання програмних засобів

- ✓ підтримувати ревізії і тестування програмного продукту, що проводяться покупцем в процесі приймання
- ✓ комплектувати і поставляти програмний продукт
- ✓ забезпечити початкове і продовжене навчання

Далі

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ТЕХНІЧНІ ПРОЦЕСИ. ФУНКЦІОНУВАННЯ ПРОГРАМНИХ ЗАСОБІВ

МЕТА

застосування програмного продукту в призначеному для нього середовищі і забезпечення підтримки замовників програмного продукту

ВИХІДНІ РЕЗУЛЬТАТИ



визначається стратегія функціонування



визначаються і оцінюються умови коректного функціонування програмних засобів в призначеному для них середовищі



програмні засоби тестуються і налаштовуються в призначеному для них середовищі



програмні засоби функціонують в призначеному для них середовищі



забезпечуються сприяння та консультації замовникам програмних продуктів відповідно до умов угоди

ДІЯЛЬНІСТЬ І ЗАВДАННЯ

✓	Підготовка до функціонування
✓	Активізація і контроль функціонування
✓	Використання за призначенням
✓	Підтримка замовника
✓	Рішення проблем функціонування

Далі

ПРОЦЕСИ В КОНТЕКСТІ СИСТЕМИ. ТЕХНІЧНІ ПРОЦЕСИ

Супровід програмних засобів



Мета

забезпечення ефективної за витратами підтримки поставлюваного програмного продукту

Діяльність і завдання

Вихідні результати

Припинення застосування програмних засобів



Мета

забезпечення завершення існування системного програмного об'єкта

Діяльність і завдання

Вихідні результати

ЛЕКЦІЯ 10

ПРОЦЕСИ ЖИТТЄВОГО ЦИКЛУ ПРОДУКТУ

Мета – знати процеси життєвого циклу програмного забезпечення, його інформаційний контент.

ПЛАН

1. Процеси угоди
2. Процеси організаційного забезпечення проекту
3. Процеси проекту
4. Технічні процеси
5. Процеси реалізації програмного забезпечення
6. Процеси підтримки та повторного застосування програмного забезпечення

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Контент життєвого циклу інформаційної продукції (документації)

ЛІТЕРАТУРА

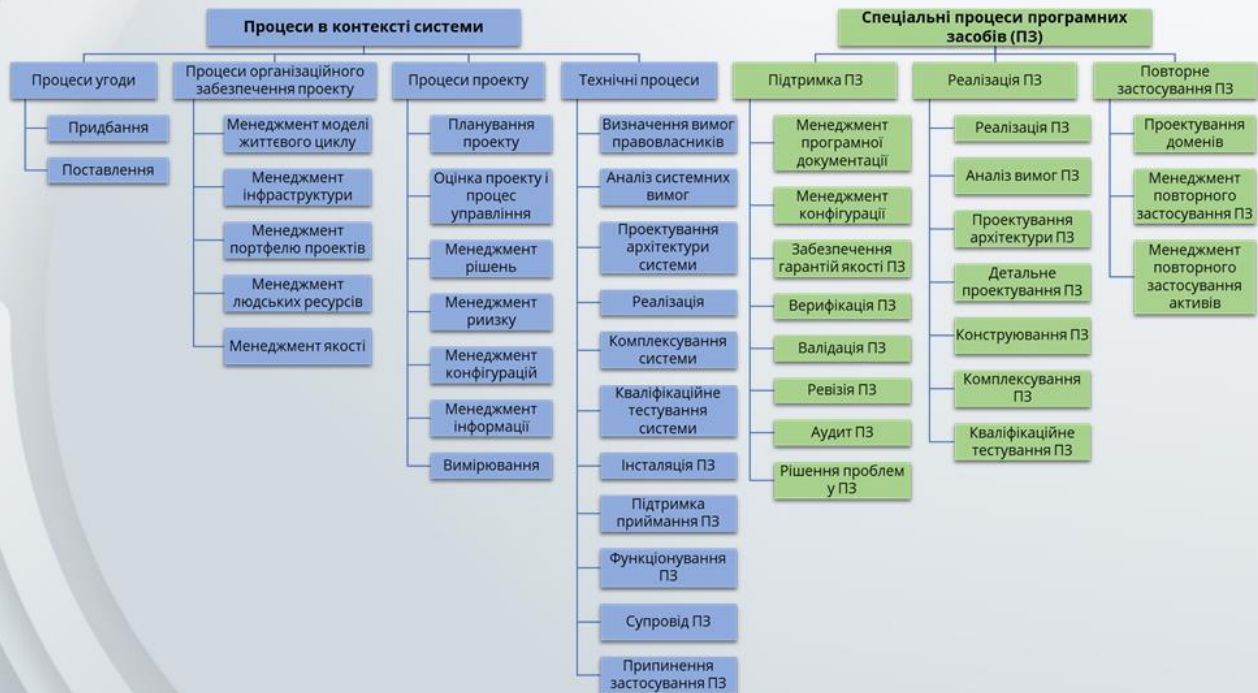
1. ISO/IEC/IEEE 12207:2017 Systems and software engineering — Software life cycle processes
2. ISO/IEC TR 24748-3:2011 Systems and software engineering — Life cycle management — Part 3: Guide to the application of ISO/IEC 12207 (Software life cycle processes)

КОНТРОЛЬНІ ПИТАННЯ

1. З яких процесів складається життєвий цикл програмного продукту?
2. З яких етапів складається процес підтримки програмного забезпечення?
3. З яких етапів складається процес реалізації програмного забезпечення?
4. З яких етапів складається процес повторного застосування програмного забезпечення?
5. Які основні атрибути опису процесу життєвого циклу продукту / системи?

ОПОРНИЙ КОНСПЕКТ

ГРУПИ ПРОЦЕСІВ ЖИТТЕВОГО ЦИКЛУ



СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ РЕАЛІЗАЦІЇ. РЕАЛІЗАЦІЯ

Мета

- створення заданих елементів системи, виконаних у вигляді програмних продуктів або послуг

Вихідні результати

- визначається стратегія реалізації
- визначаються обмеження за технологією реалізації проекту
- виготовляється програмна складова частина
- програмна складова частина пакується і зберігається відповідно до угоди про її постачання

Діяльність та завдання

- Стратегія реалізації програмних засобів

встановлювати базові лінії та з'єднувати елементи конфігурації в терміни, визначені стороною, що купує, і постачальником

вибирати, адаптувати і застосовувати документально оформлені стандарти, методи, інструментарій та мови програмування

документувати результати відповідно до процесу менеджменту програмної документації

виконувати підтримку процесів відповідно до контракту

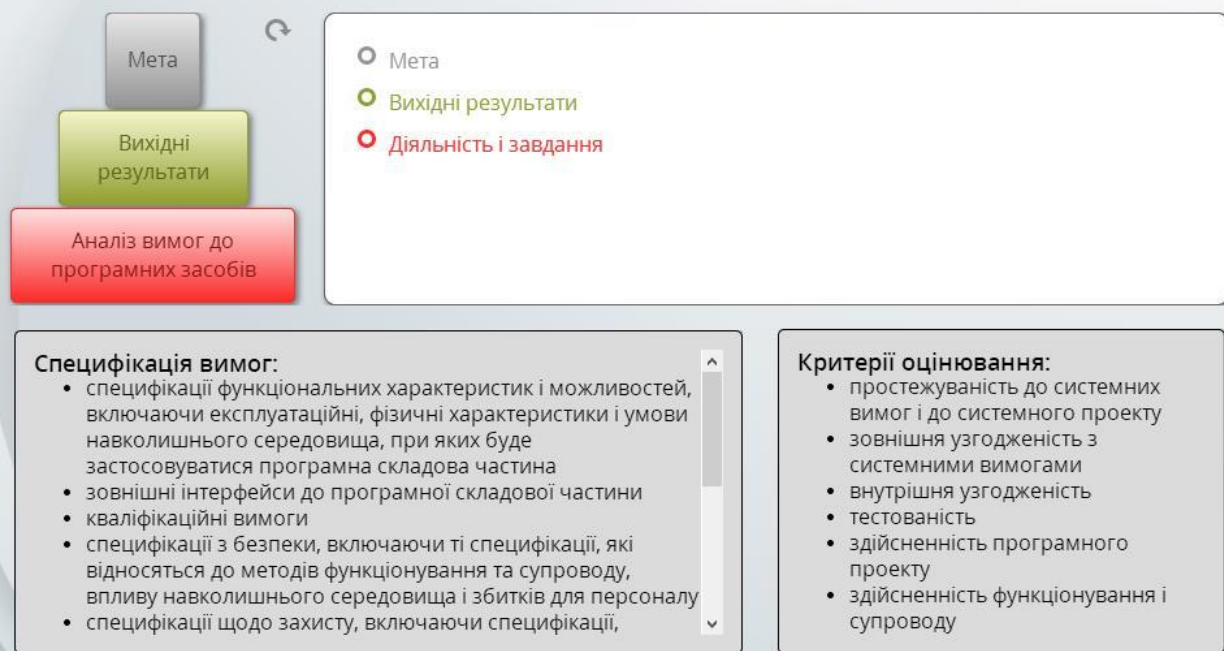
визначити чи вибрати модель життєвого циклу

розробляти плани реалізації

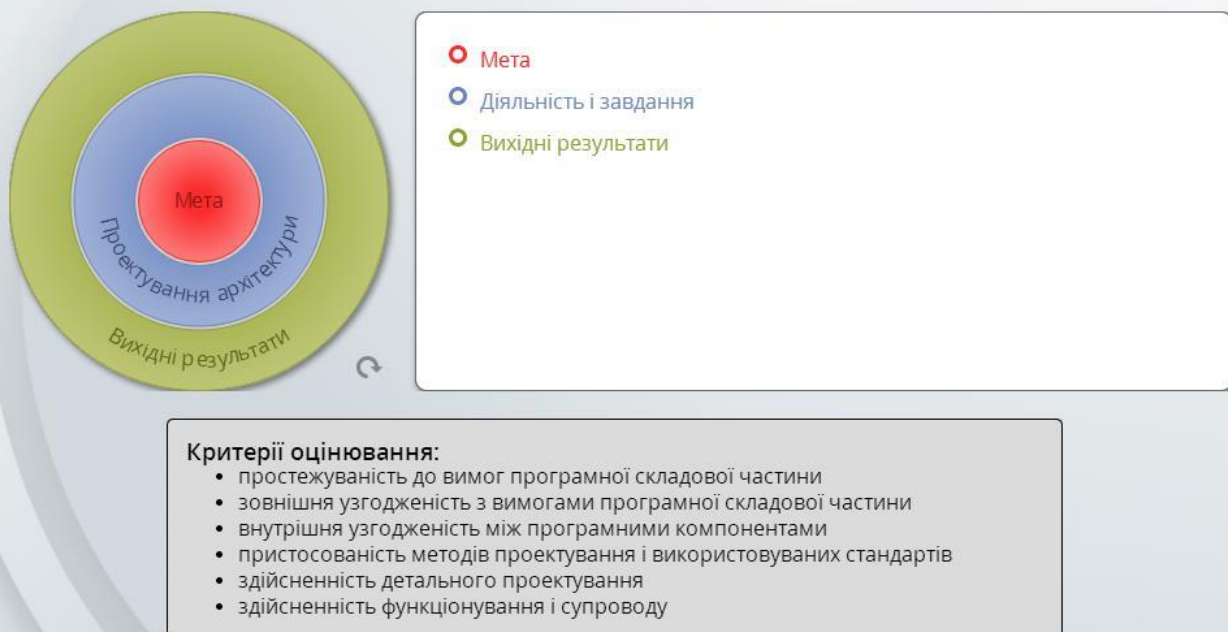
передавати результати в процес менеджменту конфігурації програмних засобів і виконувати управління змінами відповідно до нього

документувати, вирішувати проблеми і знімати невідповідності, знайдені в програмних продуктах і завданнях відповідно до процесу вирішення проблем в програмних засобах

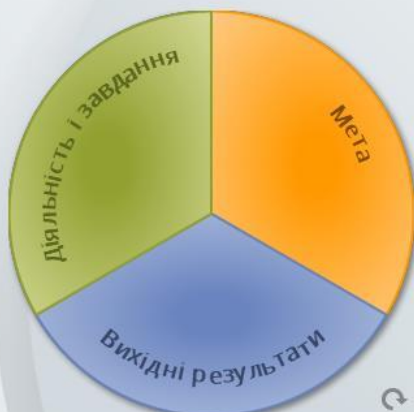
СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ РЕАЛІЗАЦІЇ. АНАЛІЗ ВИМОГ



СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ РЕАЛІЗАЦІЇ. ПРОЕКТУВАННЯ АРХІТЕКТУРИ



СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ РЕАЛІЗАЦІЇ. ДЕТАЛЬНЕ ПРОЕКТУВАННЯ

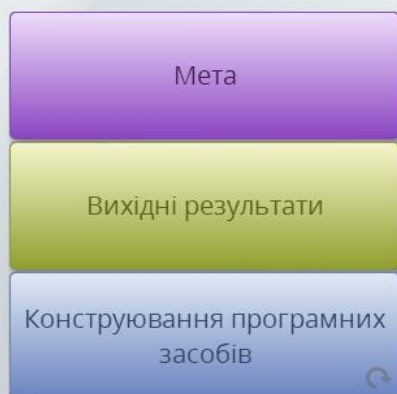


- Мета
- Вихідні результати
- Діяльність і завдання

Критерії оцінювання:

- простежуваність до вимог програмної складової частини
- зовнішня узгодженість з архітектурним проектом
- внутрішня узгодженість між програмними компонентами і програмними блоками
- відповідність методів проектування і використовуваних стандартів
- здійсненність тестування
- здійсненність функціонування і супроводу

СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ РЕАЛІЗАЦІЇ. КОНСТРУЮВАННЯ



- Мета
- Вихідні результати
- Діяльність і завдання

Критерії оцінювання:

- простежуваність до вимог і проекту програмних елементів
- зовнішня узгодженість з вимогами і проектом для програмних складових частин
- внутрішня узгодженість між вимогами до блоків
- тестове покриття блоків
- відповідність методів кодування і використовуваних стандартів
- здійсненність комплексування і тестування програмних засобів
- здійсненність функціонування і супроводу

СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ РЕАЛІЗАЦІЇ

Комплексування



Мета

об'єднання програмних блоків і компонентів, створення інтегрованих програмних елементів, узгоджених з проектом програмних засобів, які демонструють, що функціональні і нефункціональні вимоги до програмних засобів задовольняються на повністю укомплектованій або еквівалентній їй операційній платформі

Діяльність і завдання

Вихідні результати

Кваліфікаційне тестування



Мета

Діяльність і завдання

Вихідні результати

підтвердження того, що комплектуваний програмний продукт відповідає встановленим вимогам

СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ ПІДТРИМКИ. МЕНЕДЖМЕНТ ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ

МЕТА

розробка і супровід зареєстрованої інформації з програмних засобів, створеної деяким процесом

ВИХІДНІ РЕЗУЛЬТАТИ

1 розробляється стратегія ідентифікації документації, яка реалізується протягом життєвого циклу програмного продукту або послуги

2 розробляється, яка реалізується і процесом

3 розробляється, яка реалізується і процесом

4 розробляється, яка реалізується і процесом

5 розробляється, яка реалізується і процесом

6 розробляється, яка реалізується і процесом

ДІЯЛЬНІСТЬ І ЗАВДАННЯ

- ☒ Реалізація процесу
- ☒ Проектування і розробка відповідно до придатних стандартів
- ☒ Виробництво документації
- ☒ Супровід документації

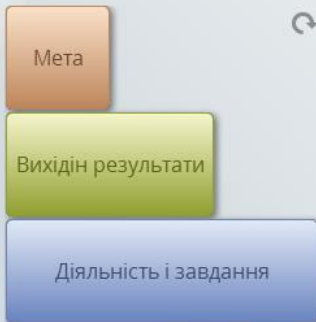
Далі

Ідентифікована документація містить:

- заголовок або назва
- цілі і зміст
- коло користувачів, яким вона призначена
- процедури і відповідальність при формуванні вихідних даних, розробці, ревізіях, модифікації, затвердженні, виробництві, зберіганні, розподілі, супроводі та менеджменті конфігурації
- графіки створення проміжних і остаточних версій

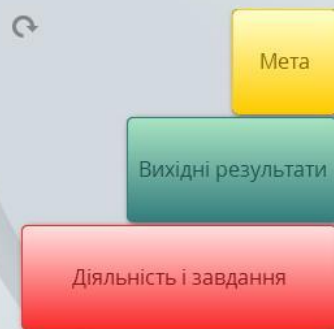
СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ ПІДТРИМКИ

МЕНЕДЖМЕНТ
КОНФІГУРАЦІЇ



- Мета
- Вихідні результати
- Діяльність і завдання

ЗАБЕЗПЕЧЕННЯ ГАРАНТІЇ
ЯКОСТІ



- Мета
- Вихідні результати
- Діяльність і завдання

СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ ПІДТРИМКИ. ВЕРИФІКАЦІЯ

МЕТА

підтвердження того, що кожен програмний робочий продукт і (або) послуга процесу або проекту належним чином відображають задані вимоги

ВІХІДНІ РЕЗУЛЬТАТИ

ДІЯЛЬНІСТЬ І ЗАВДАННЯ

- 1 розробляється і здійснюється стратегія верифікації
- 2 верифікації всіх необхідних програмних робочих продуктів
- 3 визначаються і реєструються дефекти
- 4 результати верифікації стають доступними замовнику та іншим зацікавленим сторонам

1. Реалізація процесу

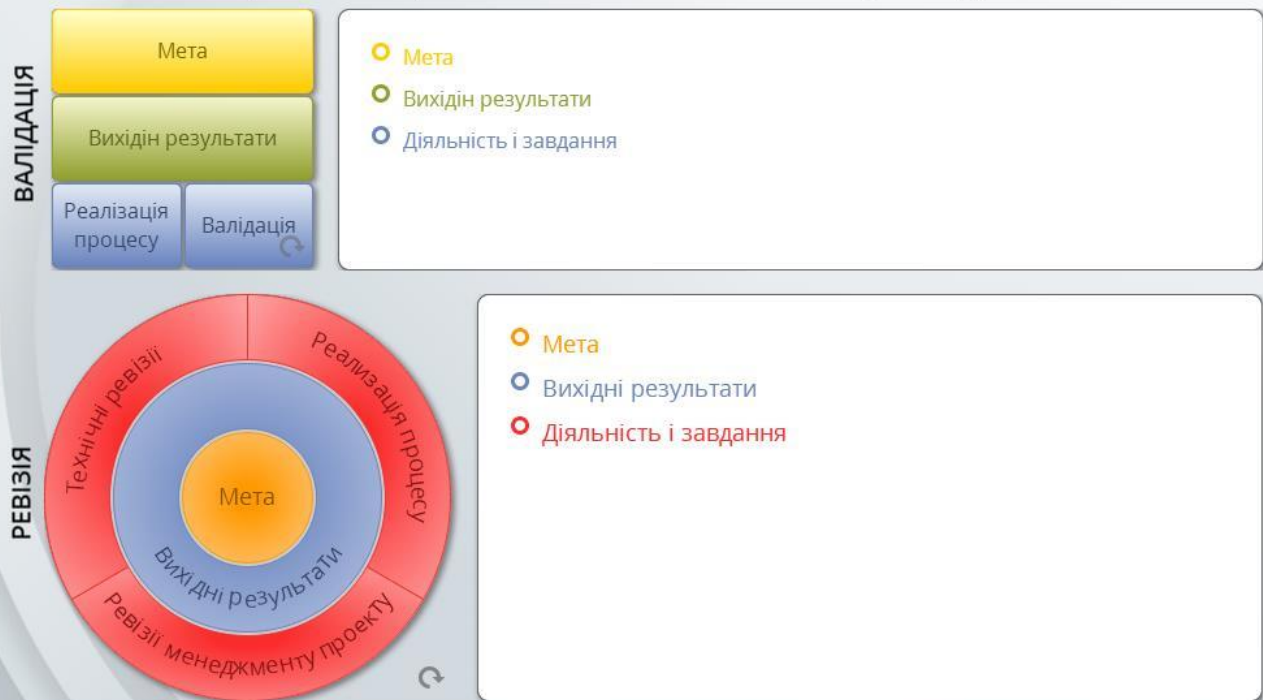
- визначити умови
- проаналізувати вимоги проекту на критичність
- встановити процес верифікації
- вибрати кваліфіковану організацію (при незалежній верифікації)
- визначити програмні продукти, що вимагають верифікації, і кінцеві цілі дій протягом життєвого циклу

2. Верифікація

- ✓ [вимог](#)
- ✓ [проекту](#)
- ✓ [коду](#)
- ✓ [комплексування](#)
- ✓ [документації](#)

Далі

СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ ПІДТРИМКИ



СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ ПІДТРИМКИ. АУДИТ

МЕТА

незалежне визначення відповідності обраних продуктів і процесів вимогам, планам і угодам

ВИХІДНІ РЕЗУЛЬТАТИ

- розробляється і здійснюється стратегія аудиту
- згідно зі стратегією аудиту визначається відповідність відібраних робочих програмних продуктів і (або) послуг або процесів вимогам, планам і угодам
- аудити проводяться відповідними незалежними сторонами
- проблеми, виявлені в процесі аудиту, ідентифікуються, доводяться до відома відповідальних за коригувальні дії і потім

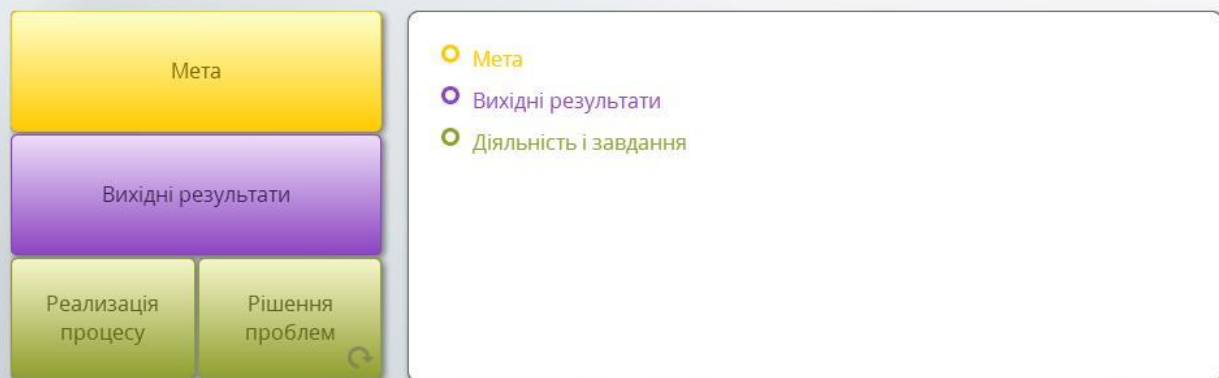
Діяльність і завданч

Реалізація процесу

узгоджувати за кожним аудитом: склад перевірюваних програмних продуктів (і результати діяльності); область поширення і процедури аудиту; вихідні та підсумкові критерії проведення аудиту

Аудит програмних засобів

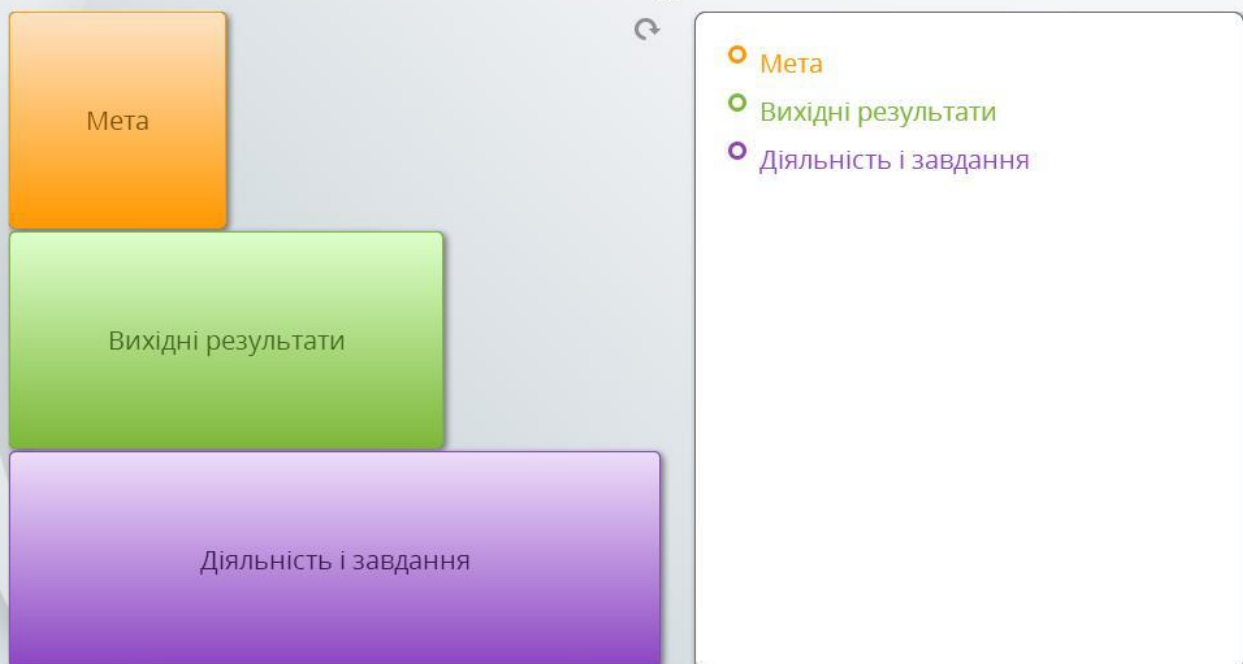
СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ ПІДТРИМКИ. ВИРІШЕННЯ ПРОБЛЕМ



Замкнута петля повинна гарантувати, що:

- про всі виявлені проблеми негайно повідомляється, і вони вводяться в процес вирішення проблем
- з цих проблем ініціюються необхідні дії
- відповідні сторони, як прийнято, інформують про існування проблем
- причини встановлюються, аналізуються і, якщо можливо, усуваються
- досягають рішення і їх поширення
- стан проблеми відстежується і відбивається в звітах
- звіти про проблеми супроводжують, як обумовлено в контракті

СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ ПОВТОРНОГО ЗАСТОСУВАННЯ. ПРОЕКТУВАННЯ ДОМЕНІВ



СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ ПОВТОРНОГО ЗАСТОСУВАННЯ. МЕНЕДЖМЕНТ ПОВТОРНОГО ЗАСТОСУВАННЯ АКТИВІВ



- Мета
- Вихідні результати
- Діяльність і завдання

- здійснювати і підтримувати механізм зберігання і пошуку активів
- розробляти, документально оформляти і супроводжувати схему класифікації активів
- проводити ревізії механізму зберігання і пошуку активів

- кожен актив повинен бути оцінений на основі критеріїв приймання та сертифікації активу
- кожен прийнятий актив повинен бути доступний для повторного використання через механізм зберігання і пошуку активів
- актив повинен бути класифікований
- виконувати управління конфігурацією для активів
- відслідковувати кожне повторне застосування активу і повідомляти інформацію розробнику доменів про поточні повторні використання активу
- скласти заявки на модифікації активів і звіти про проблеми

СПЕЦІАЛЬНІ ПРОЦЕСИ ПРОГРАМНИХ ЗАСОБІВ. ПРОЦЕСИ ПОВТОРНОГО ЗАСТОСУВАННЯ. МЕНЕДЖМЕНТ ПОВТОРНОГО ЗАСТОСУВАННЯ ПРОГРАМ

Мене́джмент повторного застосування



Мета

Діяльність і
завдання

Вихідні
результати

планування, створення, керівництво, управління і моніторинг повторного застосування програм в організації при систематичному використанні можливостей повторного застосування

Діяльність і завдання



Ініціація

визначити стратегію, спонсорів, учасників, процеси реалізації та підтримки повторного застосування

Ідентифікація домену

Планування

Виконання і управління

ЛЕКЦІЯ 11

МОДЕЛІ ЖИТТЄВОГО ЦИКЛУ

Мета – знати види моделей життєвого циклу програмного забезпечення, їх переваги та недоліки; вміти обирати модель життєвого циклу продукту у залежності від умов його розробки та застосування.

ПЛАН

1. Каскадна модель
2. Інкрементна модель
3. Спиральна модель
4. Еволюційна модель

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. V-модель (розробка через тестування)
2. Гнучка модель

ЛІТЕРАТУРА

1. Липаев В. В. Сертификация программных средств: Учебник / В. В. Липаев. – М.: Синтег, 2009. – 336 с.
2. ISO/IEC TR 24774:2010 Systems and software engineering — Life cycle management — Guidelines for process description
3. ISO/IEC/IEEE 15289:2019 Systems and software engineering — Content of life-cycle information items (documentation)
4. ISO/IEC/IEEE 15939:2017 Systems and software engineering — Measurement process

КОНТРОЛЬНІ ПИТАННЯ

1. Які найпоширеніші соделі життєвого циклу програмного забезпечення?
2. Які переваги та недоліки каскадної моделі?
3. За якою моделлю можливе поетапне впровадження продукту?
4. Яка модель вимагає аналізування ризиків?
5. Як обрати необхідну модель життєвого циклу програмного забезпечення?

ОПОРНИЙ КОНСПЕКТ

КАСКАДНА МОДЕЛЬ

Розробка вимог

Проектування

Реалізація

Тестування та
налагодження

Експлуатація та супровід

СФЕРА
ЗАСТОСУ-
ВАННЯ

1

при розробці проектів з чіткими, незмінними протягом життєвого циклу вимогами, зрозумілими реалізацією і технічними методиками

2

при виконанні великих проектів, в яких задіяно кілька великих команд розробників

3

при розробці проекту, орієнтованого на побудову системи або продукту такого ж типу, як вже розроблялися розробниками раніше

4

при розробці проекту, пов'язаного зі створенням і випуском нової версії вже існуючого продукту або системи

5

при розробці проекту, пов'язаного з перенесенням вже існуючого продукту або системи на нову платформу

ПЕРЕ- ВАГИ

i

Недоліки

складність чіткого формулювання вимог і неможливість їх динамічної зміни поки йде повний життєвий цикл

1 2 3 4 5 6 7 8 9 10

01

стабільність вимог протягом усього процесу розробки

02

конкретність і зрозумілість кроків моделі і простота її застосування

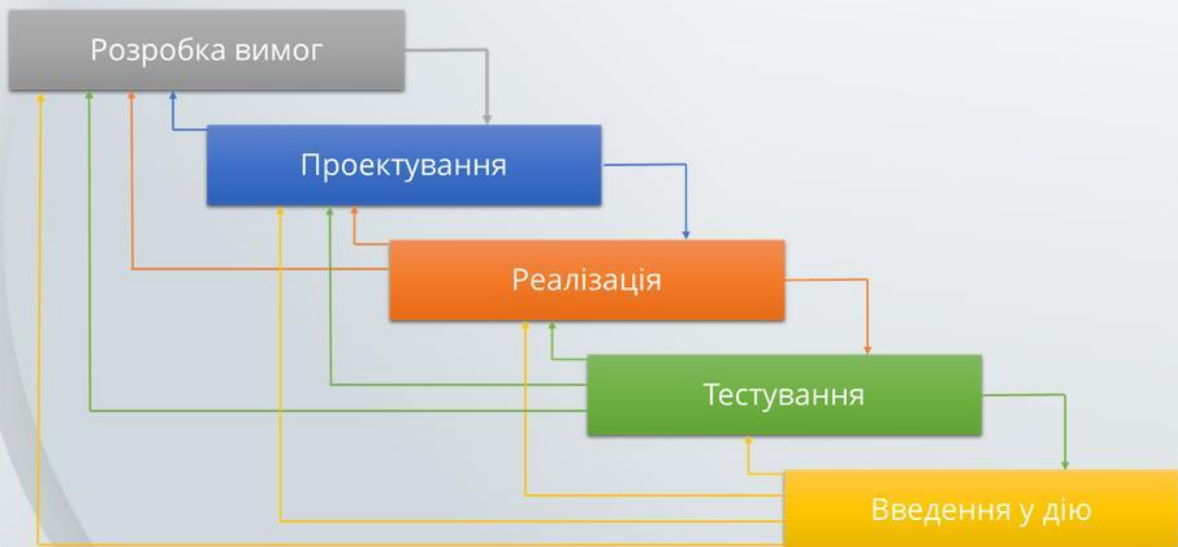
03

на кожній стадії формується закінчений набір проектної документації, який відповідає критеріям повноти і узгодженості

04

виконувані в логічній послідовності етапи робіт дозволяють планувати терміни завершення всіх робіт і ресурси

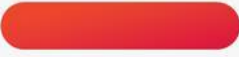
ІНКРЕМЕНТНА МОДЕЛЬ



ПРИЧИНИ ВИКОРИСТАННЯ



відсутність у замовника можливості відразу профінансувати весь дорогий проект



відсутність у розробника необхідних ресурсів для реалізації складного проекту в стислі терміни



вимога поетапного впровадження і освоєння продукту кінцевими користувачами



Поетапне впровадження продукту

Впровадження всієї системи відразу може викликати у її користувачів неприйняття і тільки "загальмувати" процес переходу на нові технології. Образно кажучи, вони можуть просто "не переварити великий шматок, тому його треба подрібнити і давати по частинах".



ПЕРЕВАГИ

витрати, які отримують в зв'язку зі зміною вимог користувачів, зменшуються, повторний аналіз і сукупність документації значно скорочуються в порівнянні з каскадною моделлю

легше отримати відгуки від клієнта про виконану роботу - клієнти можуть озвучити свої коментарі щодо готових частин і можуть бачити, що вже зроблено, тому що перші частини системи є прототипом системи в цілому

у клієнта є можливість швидко отримати і освоїти програмне забезпечення - вони можуть отримати реальні переваги від системи раніше, ніж це було б можливо з каскадною моделлю

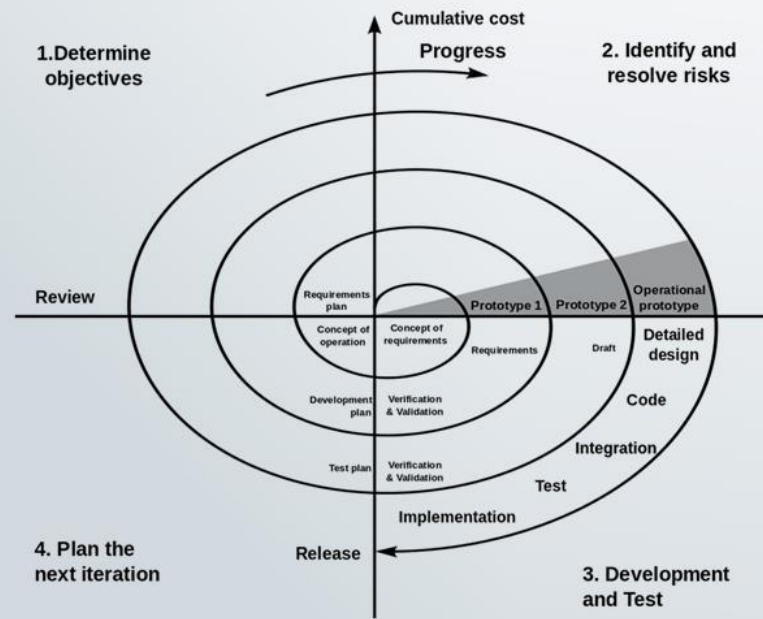
НЕДОЛІКИ

менеджери повинні постійно вимірювати прогрес процесу, в разі швидкої розробки не варто створювати документи для кожної мінімальної зміни версії

структура системи має тенденцію до погіршення при додаванні нових компонентів, щоб уникнути цього потрібен додатковий час і гроші на рефакторинг; погана структура робить програмне забезпечення складним і дорогим для подальших змін, а перерваний життєвий цикл призводить до ще більших втрат



СПІРАЛЬНА МОДЕЛЬ



СФЕРА ЗАСТОСУВАННЯ

1 при розробці проектів, що використовують нові технології

2 при розробці нової серії продуктів або систем

3 при розробці проектів з очікуваними істотними змінами або доповненнями вимог

4 для виконання довгострокових проектів

5 при розробці проектів, які потребують демонстрації якості і версій системи або продукту через короткий період часу

6 при розробці проектів, для яких необхідний підрахунок витрат, пов'язаних з оцінкою та рішенням ризиків



ПЕРЕВАГИ

- 1 дозволяє швидше показати користувачам системи працездатний продукт, тим самим, активізуючи процес уточнення і доповнення вимог
- 2 допускає зміну вимог при розробці програмного забезпечення, що характерно для більшості розробок, в тому числі і типових
- 3 в моделі передбачена можливість гнучкого проектування, оскільки в ній втілені переваги каскадної моделі, і в той же час дозволені ітерації за всіма фазами цієї ж моделі
- 4 дозволяє користувачам активно брати участь при плануванні, аналізі ризиків, розробці, а також при виконанні оціночних дій
- 5 дозволяє отримати більш надійну і стійку систему, у міру розвитку продукту помилки і слабкі місця виявляються і виправляються на кожній ітерації
- 6 зменшуються ризики замовника, він може з мінімальними для себе фінансовими втратами завершити розвиток безперспективного проекту
- 7 зворотний зв'язок у напрямку від користувачів до розробників виконується з високою частотою і на ранніх етапах, що забезпечує створення необхідного продукту високої якості

НЕДОЛІКИ

велика кількість проміжних циклів може привести до необхідності обробки додаткової документації

використання моделі може виявитися дорогим і навіть неприпустимим за коштами, тому що час, витрачений на планування, повторне визначення цілей, виконання аналізу ризиків та прототипування, може бути надмірним

можуть виникнути труднощі при визначенні цілей і стадій, які вказують на готовність продовжувати процес розробки на наступний

якщо проект має низький ступінь ризику або невеликі розміри, модель може виявитися дорогою, оцінка ризиків після проходження кожної спіралі пов'язана з великими витратами

життєвий цикл моделі має ускладнену структуру, тому може бути ускладнене її застосування розробниками, менеджерами і замовниками

спіраль може тривати до нескінченності, оскільки кожна відповідна реакція замовника на створену версію може породжувати новий цикл, що віддаляє закінчення роботи над проектом

Недоліки спіральної моделі

Основна проблема спірального циклу - визначення моменту переходу на наступний етап. Для її рішення вводяться часові обмеження на кожен з етапів життєвого циклу і перехід здійснюється відповідно до плану, навіть якщо не вся запланована робота закінчена. Планування проводиться на основі статистичних даних, отриманих в попередніх проектах і особистого досвіду розробників.

ЕВОЛЮЦІЙНА МОДЕЛЬ



НЕДОЛІКИ

реалізація всіх функцій системи одночасно може привести до громіздкості

обмежені людські ресурси зайняті розробкою протягом тривалого часу

ПЕРЕВАГИ

швидка реалізація деяких функціональних можливостей системи і їх апробація

використання проміжного продукту в наступному прототипі

зворотний зв'язок встановлюється із замовником для уточнення функціональних вимог

виділення окремих функціональних частин для реалізації їх у вигляді прототипу

можливість збільшення фінансування системи

спрощення внесення змін у зв'язку із заміною окремої функції

КРИТЕРІЙ ВИБОРУ МОДЕЛІ ЖИТТЄВОГО ЦИКЛУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



ЛЕКЦІЯ 12 КЕРУВАННЯ РИЗИКАМИ

Мета – знати модель та процеси менеджменту ризику, особливості його застосування; вміти складати рекомендації щодо розробки та впровадження менеджменту ризику на підприємстві з розробки програмного забезпечення.

ПЛАН

1. Модель процесу менеджменту ризику
2. Планування і впровадження менеджменту ризику
3. Управління профілем ризику проекту
4. Аналіз, обробка та моніторинг ризику
5. Оцінка процесу менеджменту ризику

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Особливості промислового застосування менеджменту ризику
2. Рекомендації щодо розробки та впровадження менеджменту ризику

ЛІТЕРАТУРА

1. ISO/IEC 16085:2006 Systems and software engineering — Life cycle processes — Risk management

КОНТРОЛЬНІ ПИТАННЯ

1. З яких процесів складається модель менеджменту ризику?
2. З чого складається управління профілем ризику проекту?

3. Навіщо виконують моніторинг ризику?
4. Які документи складають у результаті менеджменту ризику?
5. З яких завдань складається оцінка процесу менеджменту ризику?

ОПОРНИЙ КОНСПЕКТ

МЕНЕДЖМЕНТ РИЗИКУ ЖИТТЄВОГО ЦИКЛУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

МЕТА

постійна ідентифікація, аналіз, обробка та моніторинг ризику

РЕЗУЛЬТАТ

✓	визначення області застосування менеджменту ризику
✓	визначення та впровадження відповідних стратегій менеджменту ризику
✓	ідентифікація ризику на всіх стадіях реалізації проекту
✓	проведення аналізу ризику та визначення пріоритетних областей вкладення ресурсів, необхідних для моніторингу ризику
✓	визначення, застосування та оцінка ризику для визначення зміни статусу ризику і його моніторингу
✓	проведення відповідної обробки ризику, необхідної для коригування і запобігання впливу ризику і заснованої на визначенні пріоритетів, ймовірності та наслідків

Далі

МОДЕЛЬ ПРОЦЕСУ МЕНЕДЖМЕНТУ РИЗИКУ



ПЛАНУВАННЯ ТА ВПРОВАДЖЕННЯ МЕНЕДЖМЕНТУ РИЗИКУ



ПОЛІТИКА В ОБЛАСТІ МЕНЕДЖМЕНТУ РИЗИКУ ЗАБЕЗПЕЧУЄ

✓	впровадження, управління та підтримку в робочому стані менеджменту ризику з боку керівництва і решти персоналу організації
✓	розподіл обов'язків з менеджменту ризику між причетними сторонами
✓	скоординовану діяльність з процесу менеджменту ризику між причетними сторонами
✓	навчання персоналу з процесу ризик-менеджменту і наявність вимог до кваліфікації персоналу в області ризику
✓	необхідну інформацію про ризик, наприклад, яким чином і як часто профіль ризику проекту повинен бути спрямований причетним сторонам для розгляду
✓	ресурси, необхідні для обробки ризику

Далі

ОПИС ПРОЦЕДУР, НЕОБХІДНИХ ДЛЯ ВПРОВАДЖЕННЯ І ФУНКЦІОНУВАННЯ ПРОЦЕСУ МЕНЕДЖМЕНТУ РИЗИКУ

✓	частота повторного аналізу і моніторингу ризику
✓	тип аналізу ризику (кількісний і / або якісний)
✓	шкала оцінки ймовірності небезпечної події та її наслідків, невизначеність вимірювань і опис наслідків
✓	типи використовуваних допустимих ризиків
✓	типи засобів вимірювальної техніки, які використовуються для простеження і моніторингу стану ризику
✓	пріоритетність обробки ризику
✓	визначення причетних сторін процесу ризик-менеджменту
✓	категорії ризику

Далі

УПРАВЛІННЯ ПРОФІЛЕМ РИЗИКУ ПРОЕКТУ



А ВИЗНАЧЕННЯ ОСОБЛИВОСТЕЙ МЕНЕДЖМЕНТУ РИЗИКУ

✓	цілі (наприклад, ключові технічні, політичні або економічні критерії виконання вимог проекту)
✓	припущення (наприклад, міркування, що виходять за межі управління проектом)
✓	обмеження (наприклад, встановлені межі проекту)

В ВСТАНОВЛЕННЯ ДОПУСТИМОГО РИЗИКУ

✓	особливості менеджменту ризику
✓	хронологічні записи про статус кожного ризику, в т. ч. ймовірність появи небезпечної події, її наслідки і допустимий ризик
✓	пріоритет кожного ризику на підставі критеріїв, представлених причетними сторонами
✓	інформація про дії з ризиком, статус його обробки

С ВСТАНОВЛЕННЯ І ПІДТРИМКА ПРОФІЛЮ РИЗИКУ ПРОЕКТУ



Д ОБМІН ІНФОРМАЦІЄЮ ПРО СТАТУС РИЗИКУ

АНАЛІЗ РИЗИКУ



1 ЦІЛІ ДІЯЛЬНОСТІ З АНАЛІЗУ РИЗИКУ

✓	ідентифікація вихідних подій, небезпек, загроз або ситуацій, які можуть викликати ризик
---	---

2 ІДЕНТИФІКАЦІЯ РИЗИКУ

✓	оцінка ймовірності виникнення небезпечних подій і їх наслідків для кожного ризику і середнього часу виникнення ризику
---	---

3 КІЛЬКІСНА ОЦІНКА РИЗИКУ

✓	перевірка кожного індивідуального ризику або їх певної комбінації на відповідність допустимому ризику, вибір альтернативних варіантів обробки ризику в разі, коли його значення вище допустимого, надання рекомендацій для обробки ризику на основі обраних пріоритетів
---	---

4 ПЕРЕВІРКА ПРИПУСТИМОСТІ РИЗИКУ

Далі

ОБРОБКА РИЗИКУ

включає в себе вибір, планування, моніторинг та управління діями для зниження експозиції ризику

01

02

03



ЦІЛІ ОБРОБКИ РИЗИКУ

- а) визначення допустимості ризику для причетних сторін
- б) вжиття необхідних заходів і дій для зниження ризику при перевищенні ним допустимого рівня

ВИБІР ОБРОБКИ РИЗИКУ

рекомендовані альтернативні методи обробки ризику

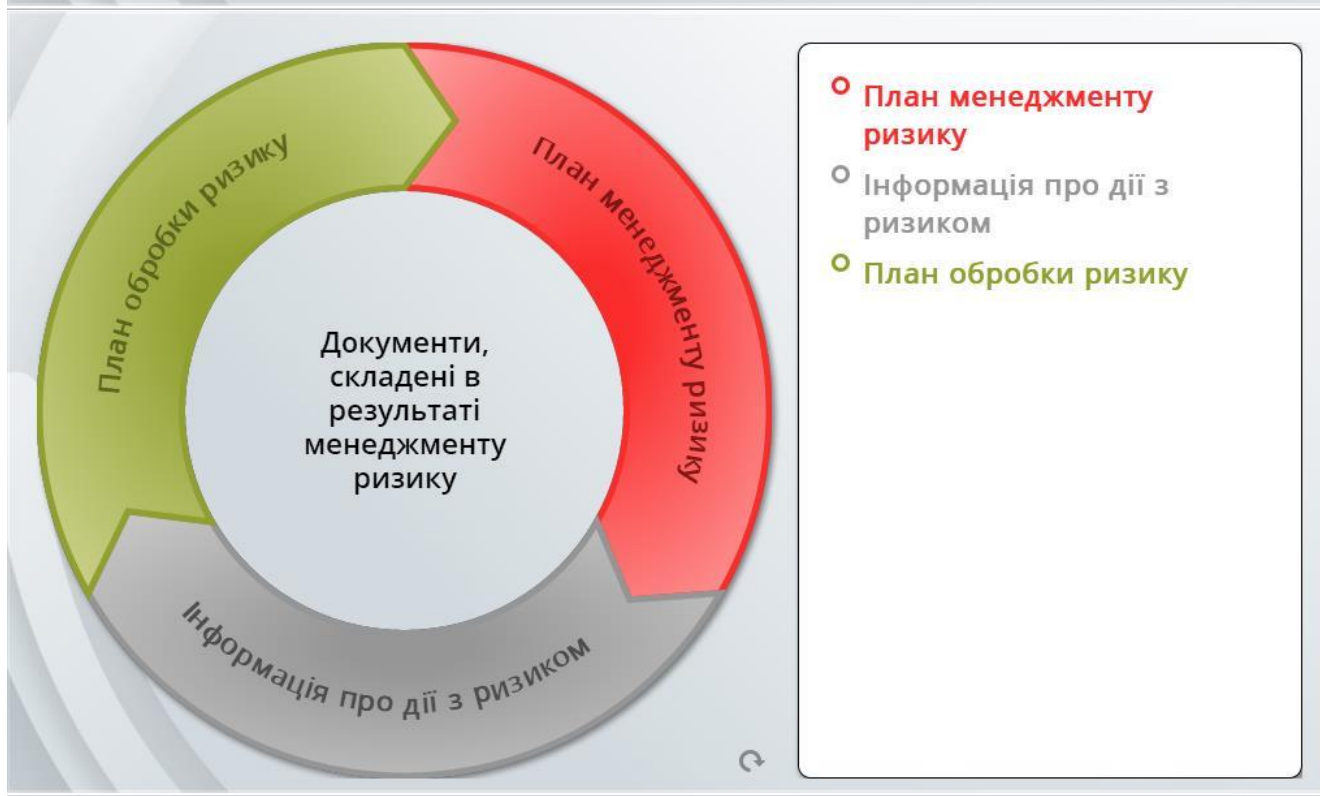
ПЛАНУВАННЯ ТА ВИКОНАННЯ ОБРОБКИ РИЗИКУ

з використанням або без використання ISO 12207 та ISO 15288



✓	аналіз і актуалізація індивідуального ризику і особливостей менеджменту ризику
✓	оцінка результативності обробки ризику
✓	пошук нового ризику
Далі	





ЛЕКЦІЯ 13

ВИМОГИ ДО СУПРОВОДУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Мета – знати загальні положення супроводу програмного забезпечення, його стратегію та процеси; вміти складати план супроводу програмного забезпечення.

ПЛАН

1. Загальні положення супроводу програмного забезпечення
2. Стратегія супроводу програмного забезпечення
3. Процес супроводу програмного забезпечення

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Загальні положення методології Scrum

ЛІТЕРАТУРА

1. ISO/IEC 14764:2006 Software Engineering — Software Life Cycle Processes — Maintenance

КОНТРОЛЬНІ ПИТАННЯ

1. Які є типи супроводу програмного забезпечення?
2. Яка структура кожного процесу супроводу?
3. Які результати отримують після підготовки процесу супроводу?
4. Які завдання виконують під час здійснення модифікацій?
5. Що має містити стратегія супроводу?

ОПОРНИЙ КОНСПЕКТ



ПРОЦЕС СУПРОВОДУ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Структура кожного процесу



ПІДГОТОВКА ПРОЦЕСУ



ЗАВДАННЯ

- ✓ розробити плани і процедури супроводу
- ✓ встановити процедури розгляду пропозиції про модифікацію і звіту про проблему
- ✓ розробити і застосувати план управління конфігурацією

Далі

Плани та процедури супроводу

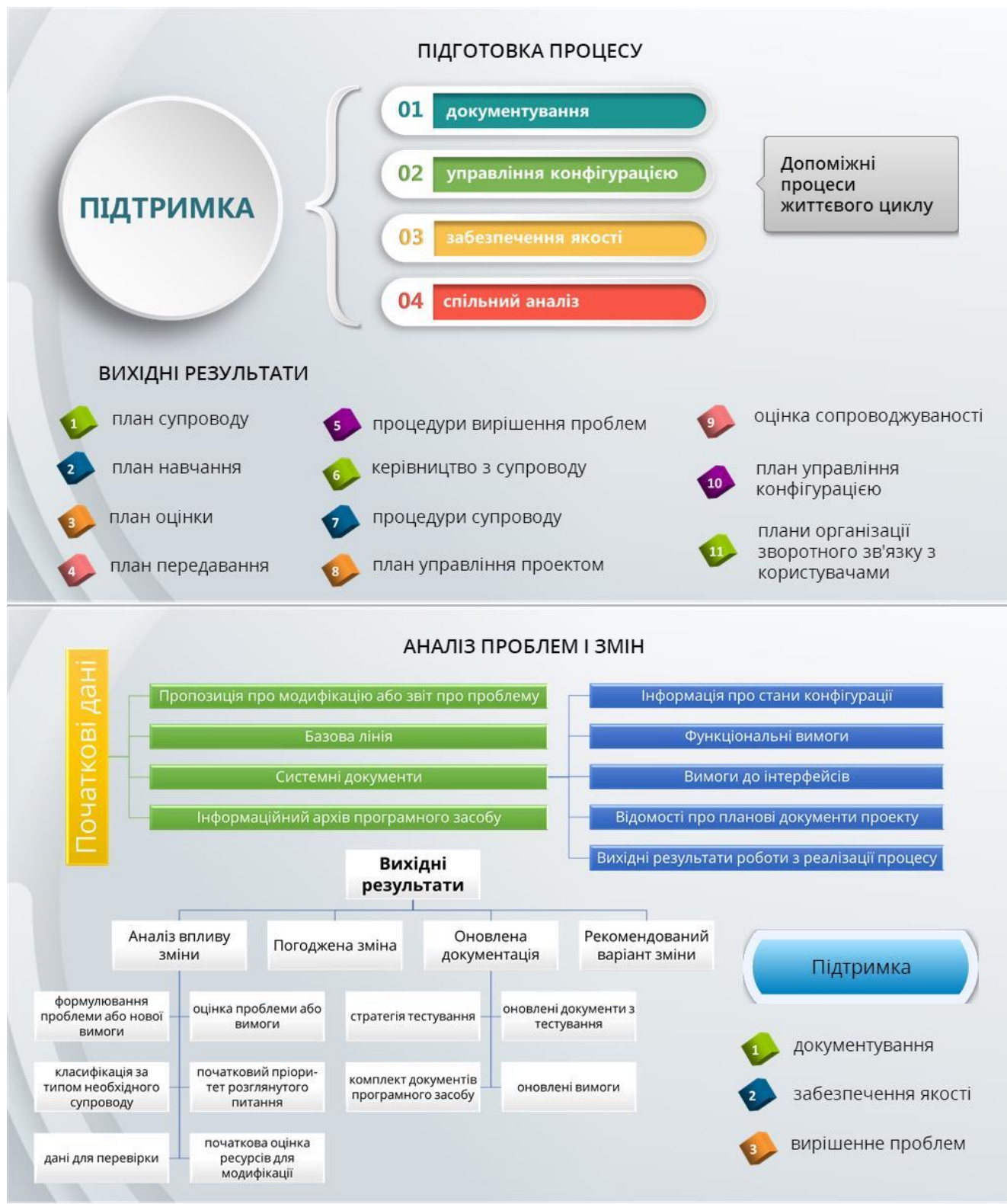
допомогти замовнику при розробці концепції супроводу

1 2 3 4 5 6 7 8 9 10 11

Пропозиція про модифікацію

розробити схему числового позначення пропозиції про модифікацію і звіту про проблему

1 2 3 4 5 6 7 8



АНАЛІЗ ПРОБЛЕМ І ЗМІН

Завдання

Аналіз пропозиції про модифікацію або звіту про проблему

Верифікація

Варіанти реалізації зміни

Документальне оформлення

Погодження

Аспекти:

- тип, наприклад: коригування, модернізація, профілактика або адаптація до нових умов
- обсяг, наприклад: розмір зміни, вартість, час на реалізацію зміни
- критичність, наприклад: вплив на продуктивність, безпеку або захист

Етапи вирішення даного завдання:

- визначити наявність відповідного персоналу, здатного реалізувати запропоновані зміни
- визначити наявність відповідного фінансування для реалізації запропонованої зміни в програмі
- визначити наявність відповідних ресурсів і ступінь впливу модифікації на виконуваний або вже реалізовані проекти (можливо, не слід реалізовувати для звіту про проблему)
- визначити, які мають враховуватися питання, пов'язані з експлуатацією, наприклад, чи впливає нереалізованість передбачуваних змін на вимоги до системних інтерфейсів, очікуваний термін служби системи, пріоритети, безпека і захист при експлуатації (можливо, не слід реалізовувати для звіту про проблему)
- визначити експлуатаційні пріоритети
- класифікувати тип супроводу
- визначити вплив на поточних і майбутніх користувачів

ЗДІЙСНЕННЯ МОДИФІКАЦІЙ



1.
 - опис системної архітектури
 - документи конкретної пропозиції про модифікацію (зміну)
 - вихідні коди програми

2.
 - звіт про аналіз впливу зміни
 - вихідні результати роботи з аналізу проблем і змін

ВИХІДНІ РЕЗУЛЬТАТИ

✓	оновлені плани і процедури тестування
✓	змінений вихідний код програми
✓	звіт про тестування
✓	показники, що характеризують внесені зміни
✓	<u>оновлена документація</u>

Далі

1. оновлені документи на зміну
2. докладний звіт про проведений аналіз
3. оновлені вимоги
4. оновлені навчальні матеріали
5. оновлені плани, процедури і звіти про тестування

ЗДІЙСНЕННЯ МОДИФІКАЦІЙ



визначити елементи, які слід змінити у наявній системі



визначити документи, що підлягають оновленню



визначити елементи конкретного інтерфейсу, які порушені даною зміною



оновити документацію програмного засобу

ЗАВДАННЯ



Аналіз



Процес розробки



Встановити і документально оформити критерії проведення випробувань, оцінки їх результатів і оцінки змінених і незмінених об'єктів (програмних модулів, компонентів і елементів конфігурації) системи



Забезпечити повноту та правильність реалізації нових і змінених вимог, щоб вихідні, незмінні, вимоги не змінилися

ПЕРЕВІРКА І ПРИЙМАННЯ ПРИ СУПРОВОДІ

ПОЧАТКОВІ
ДАНІ

ЗМІНЕНИЙ
ПРОГРАМНИЙ
ЗАСІБ

РЕЗУЛЬТАТИ
ТЕСТУВАННЯ
ВНЕСЕНОЇ ЗМІНИ

ЗАВДАННЯ

ПОГОДЖЕННЯ

ОГЛЯД

- ✓ отримання погодження за допомогою допоміжного процесу забезпечення якості
- ✓ перевірка виконання даного процесу
- ✓ підготовка пакету документів для роботи оператора
- ✓ проведення аудиту функціональної і фізичної конфігурації
- ✓ установка програмного засобу та навчання оператора
- ✓ сповіщення операторів

- ✓ відстеження реалізованої пропозиції про модифікацію або звіту про проблему від вимог до об'єкту до проекту і програмних кодів
- ✓ перевірка тестованості кодів програми
- ✓ перевірка дотримання стандартів на програмування
- ✓ перевірка того, що змінні тільки потрібні компоненти
- ✓ перевірка правильності складання нових компонентів
- ✓ контроль оновлення документів
- ✓ проведення тестування через контроль конфігурації
- ✓ проведення тестування незалежною організацією
- ✓ проведення тестування на повністю інтегрованій системі
- ✓ розробка звіту про тестування

Далі

ПЕРЕВІРКА І ПРИЙМАННЯ ПРИ СУПРОВОДІ

ПІДТРИМКА

ЗАБЕЗПЕЧЕННЯ ЯКОСТІ

ВЕРИФІКАЦІЯ

АТЕСТАЦІЯ

СПІЛЬНИЙ АНАЛІЗ

АУДИТ

ВИХІДНІ РЕЗУЛЬТАТИ

НОВА БАЗОВА ЛІНІЯ, ВКЛЮЧАЄ В СЕБЕ ПРИЙНЯТІ ЗМІНИ

ВІДХИЛЕНІ ЗМІНИ

ЗВІТ ПРО ПРИЙМАННЯ

ЗВІТИ ПРО ОГЛЯД (ПЕРЕВІРКИ) ТА АУДИТИ

ЗВІТ ПРО КВАЛІФІКАЦІЙНЕ ТЕСТУВАННЯ

ПЕРЕНЕСЕННЯ

ПОЧАТКОВІ ДАНІ

Старе середовище

Нове середовище

Стара базова лінія

Нова базова лінія

ПІДТРИМКА

Документування

Управління конфігурацією

Забезпечення якості

Верифікація

Атестація (валідація)

Спільний аналіз

Аудит

Вирішення проблем

ВИХІДНІ РЕЗУЛЬТАТИ

План перенесення

Інструментальні засоби для перенесення

Перенесений програмний продукт

Повідомлення про наміри

Повідомлення про завершення перенесення

Архівні дані

ПЕРЕНЕСЕННЯ



- Перенесення
- План перенесення
- Повідомлення про наміри
- Операції та навчання
- Повідомлення про завершення
- Підсумковий аналіз
- Архівні дані

ЗНЯТТЯ З ЕКСПЛУАТАЦІЇ

Початкові дані

- видалена базова лінія старого програмного продукту
- новий програмний продукт
- старе середовище експлуатації



П
І
Д
Т
Р
И
М
К
А



ВИХІДНІ РЕЗУЛЬТАТИ

✓	план знімання з експлуатації
✓	повідомлення про наміри щодо зняття з експлуатації
✓	результати, отримані при виконанні зняття програмного засобу з експлуатації
✓	навчений персонал
✓	знятий з експлуатації програмний продукт
✓	повідомлення про завершення зняття з експлуатації
✓	оцінки
✓	архівована базова лінія та дані

Далі

ЗНЯТТЯ З ЕКСПЛУАТАЦІЇ



01

План зняття з експлуатації



02

Повідомлення про наміри



03

Реалізація паралельної експлуатації та навчання



04

Повідомлення про завершення



05

Архівні дані



ЗАВДАННЯ

План зняття з експлуатації

- проаналізувати вимоги до зняття з

Повідомлення про наміри

Зміст:

Реалізація паралельної експлуатації та навчання

- провести обстеження абонента
- встановити обладнання

Архівні дані

- зберегти старі програмні засоби і дані, отримані при вирішенні попередніх завдань
- створити копії старих програмних засобів і даних, отриманих при вирішенні попередніх завдань
- зберігати відповідні носії в безпечному місці

СТРАТЕГІЯ СУПРОВОДУ ПРОГРАМНОГО ЗАСОБУ

Аналіз
ресурсів

План

Концепція

Політика супроводу

- Аналіз ресурсів
- **План**
- **Концепція**
- **Політика супроводу**

ЛЕКЦІЯ 14

ПОНЯТТЯ І ВИЗНАЧЕННЯ В ТЕСТУВАННІ

Мета – знати основні терміни та визначення, організацію тестування програмного забезпечення, його процеси життєвому циклі програмного забезпечення; вміти застосовувати менеджмент ризику для організації тестування програмного забезпечення.

ПЛАН

1. Основні терміни та визначення
2. Організація тестування програмного забезпечення
3. Процеси тестування у життєвому циклі програмного забезпечення
4. Тестування на базі ризиків

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Тестування і вимоги до якості

ЛІТЕРАТУРА

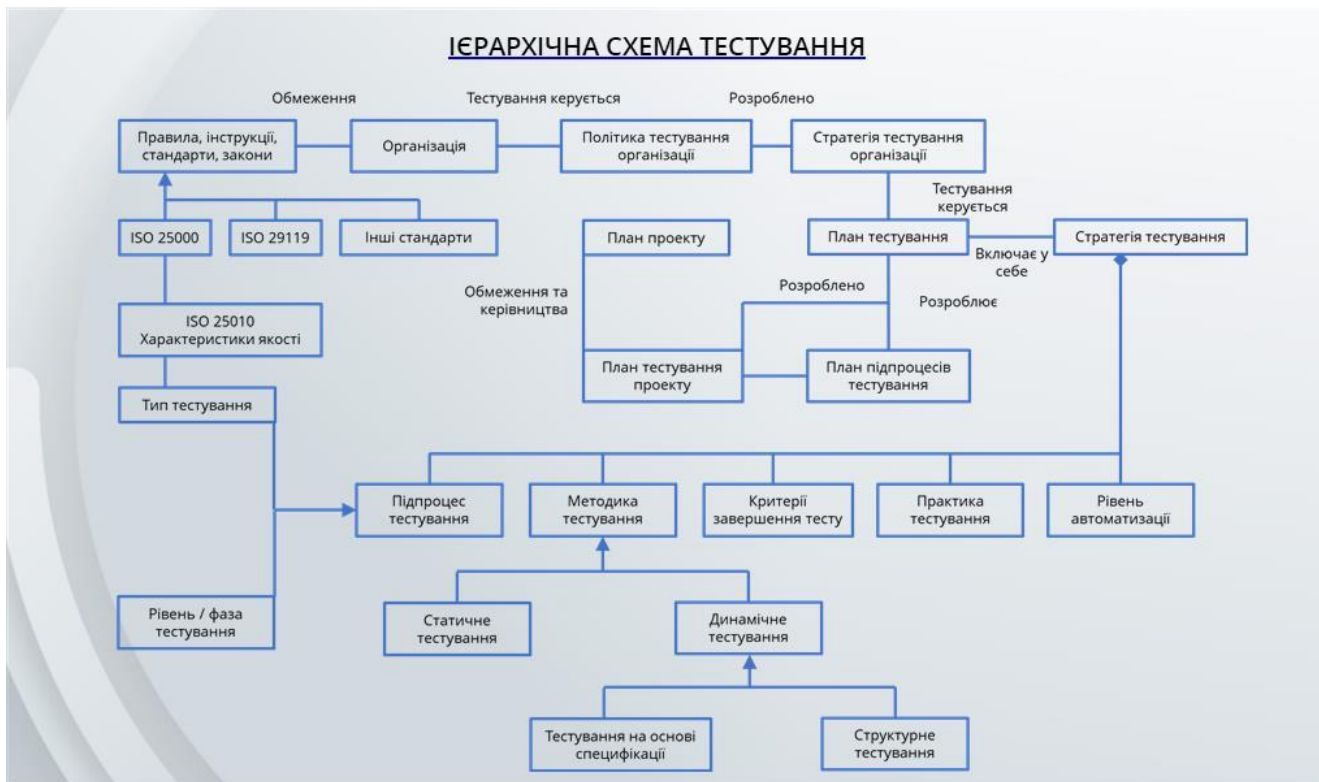
1. ISO/IEC/IEEE 29119-1:2013 Software and systems engineering — Software testing — Part 1: Concepts and definitions
2. ISO/IEC 33063:2015 Information technology — Process assessment — Process assessment model for software testing

КОНТРОЛЬНІ ПИТАННЯ

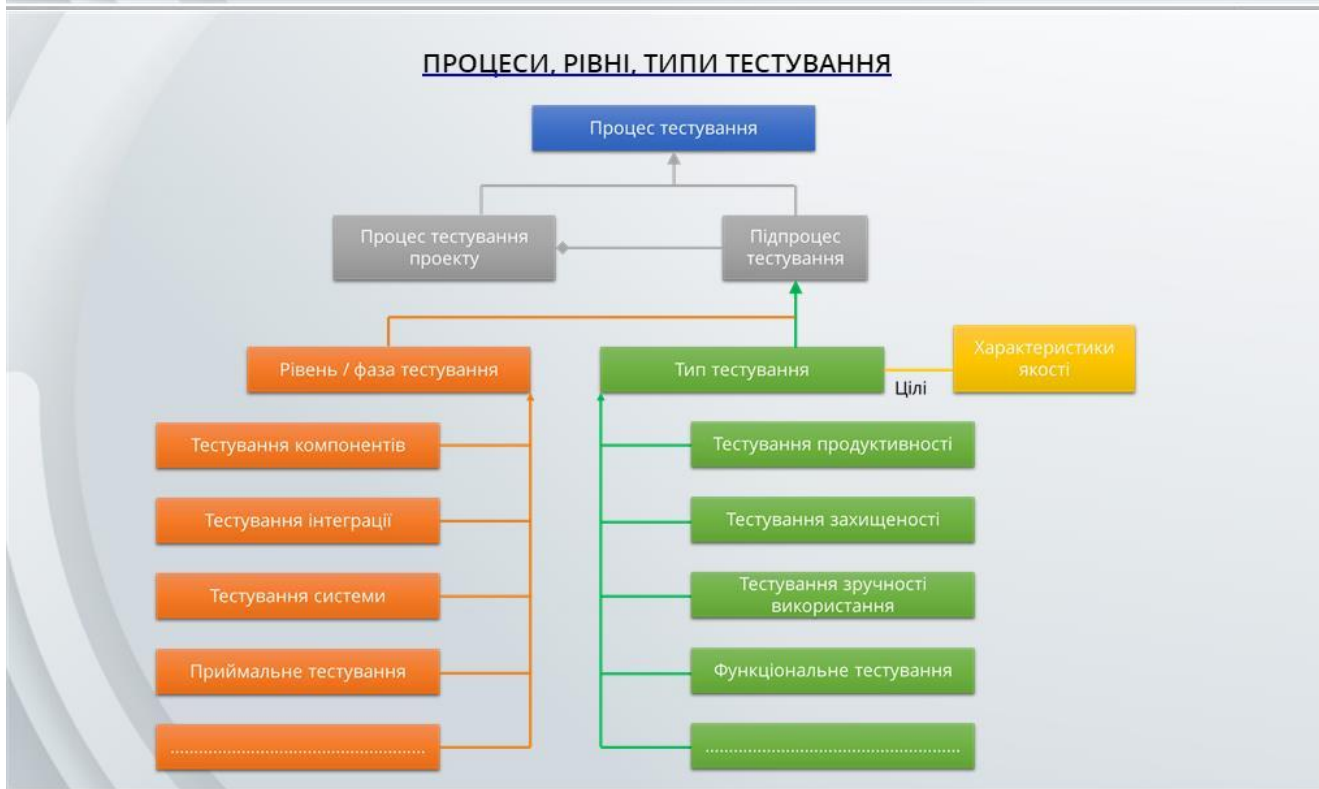
1. Які є типи та рівні тестування?
2. З яких процесів складається модель тестування?
3. Які є методики тестування?
4. Яка роль тестування у верифікації та валідації?
5. З чого складається кожний підпроцес тестування?

ОПОРНИЙ КОНСПЕКТ

ІЄРАРХІЧНА СХЕМА ТЕСТУВАННЯ



ПРОЦЕСИ, РІВНІ, ТИПИ ТЕСТУВАННЯ



МОДЕЛЬ ПРОЦЕСУ ТЕСТУВАННЯ



ВЗАЄМОЗВ'ЯЗОК ВСЬОГО ПРОЕКТУ ТА ПРОЕКТУ ТЕСТУВАННЯ



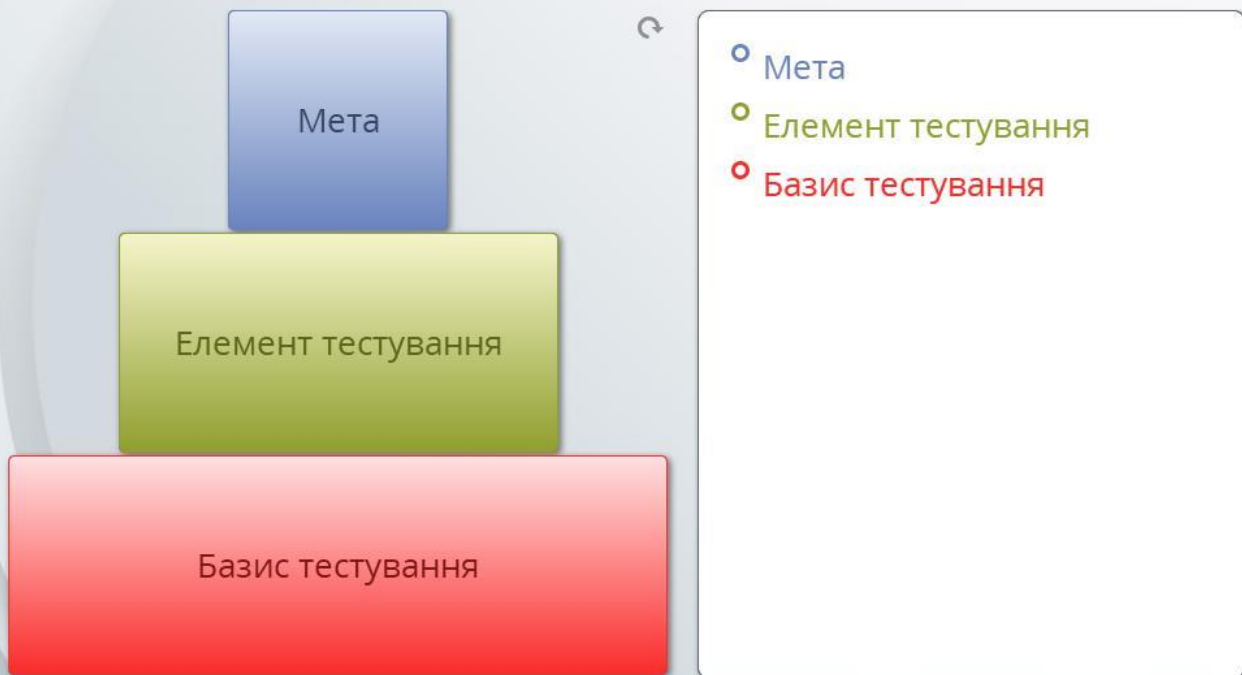
ПРОЦЕСИ ДИНАМІЧНОГО ТЕСТУВАННЯ



Динамічне тестування

Детальніше про процеси динамічного тестування див. тему 4.2.

ПІДПРОЦЕС ТЕСТУВАННЯ



ПРИКЛАД ПІДПРОЦЕСУ ТЕСТУВАННЯ В ДЕТАЛЯХ. ПІДПРОЦЕС ТЕСТУВАННЯ ПРОДУКТИВНОСТІ

МЕТА

надати інформацію про виконання вимог до продуктивності для системи

ЗАПЛАНОВАНИЙ СКЛАД ПІДПРОЦЕСУ ТЕСТУВАННЯ



[Статичне тестування 1. Документація вимог до продуктивності](#)

[Статичне тестування 2. Повнота вимог до продуктивності](#)

[Статичне тестування 3. Архітектура з точки зору продуктивності](#)

[Статичне тестування 4. Детальне проектування з точки зору продуктивності](#)

[Динамічне тестування 1. Застосовна підсистема з точки зору продуктивності](#)

[Динамічне тестування 2. Завершена система з точки зору продуктивності](#)

Динамічне тестування 2. Завершена система з точки зору продуктивності

Мета тестування:	Надати інформацію про продуктивність абсолютно інтегрованої системи
Елемент тестування:	Завершена система
Базис тестування:	Вимоги до продуктивності та можливо ідентифіковані ризики продуктивності
Докладні процеси тестування:	Розробка та реалізація тестування; встановленні та підтримка тестового середовища; виконання тесту та звітність про інциденти тестування. Деякі процедури тестування можуть бути реалізовані повторно для тестування системи
Метод(и) проектування тестування:	Застосовні методи

Методики тестування



Засноване на вимогах тестування

Модельне тестування

Математичне тестування

Засноване на досвіді тестування

Тестування з / без сценарію

Головна мета заснованого на вимогах тестування полягає в забезпеченні гарантії, що під час тестування вимоги до елементу тестування були розглянуті (тобто "покриті") для визначення того, чи відповідає елемент тестування вимогам кінцевого користувача. Це тестування використовується також для збору і надання зацікавленим сторонам іншої цінної інформації життєвого циклу, такої як ідентифіковані в елементі тестування дефекти. При застосуванні цієї практики вимоги використовуються для того, щоб надавати інформацію, необхідну для прийняття рішень з планування, проектування та реалізації тестування і виконання процесів. Слід зазначити, що засноване на вимогах тестування може бути, зокрема, використано для отримання результатів верифікації вимог, визначеної в ISO 12207.

У заснованому на вимогах тестуванні переважно використовується тестування за сценарієм. Контрольні приклади пишуться перед виконанням тесту в ході процесу розробки і реалізації тестування. Далі контрольні приклади виконуються відповідно до розкладу, визначеного в процедурі тестування. Потім проводиться аналіз результатів виконання тесту для подальшого удосконалення тестування, яке може зажадати додаткової роботи з проектування тестування і сценарію контрольного прикладу. Контрольні приклади, створювані в ході такого додаткового проектування тестування, документуються і проводяться далі в життєвому циклі виконання тесту.

Засноване на вимогах тестування може підтримуватися іншими методиками тестування, якщо ці методики допомагають продемонструвати, що вимоги тестуються належним чином. Для забезпечення гарантії виконання всіх вимог додатково до використання заснованого на вимогах тестування з метою вирішення інших ризиків можуть бути застосовані й інші методи (наприклад, тестування на базі досвіду). Незвичайною є вимога відсутності регресивних дефектів, що залишилися в поставленому продукті. Однак в цьому випадку можна застосувати дослідне тестування як спосіб вирішення цього ризику регресії.

Доцільність застосування заснованого на вимогах тестування залежить від контексту. Якщо вимоги не повні або не узгоджені, то і результуюче тестування може мати ті ж недоліки. Навіть якщо вимоги добре визначені, є небезпека того, що бюджетні обмеження й обмеження часу можуть не дозволити перевірити всі вимоги. У разі якщо вимоги



ЛЕКЦІЯ 15 ТЕСТОВІ ПРОЦЕСИ

Мета – знати багаторівневу модель процесів тестування, їх організаційний вміст; вміти обирати необхідні види тестів програмного забезпечення, складати тест-план.

ПЛАН

1. Багаторівнева модель процесу тестування
2. Організаційний процес тестування
3. Процеси менеджменту тестування
4. Процеси динамічного тестування

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Платформи для тестування проєктів з розроблення інформаційних систем
2. Вимоги щодо тестування продуктивності

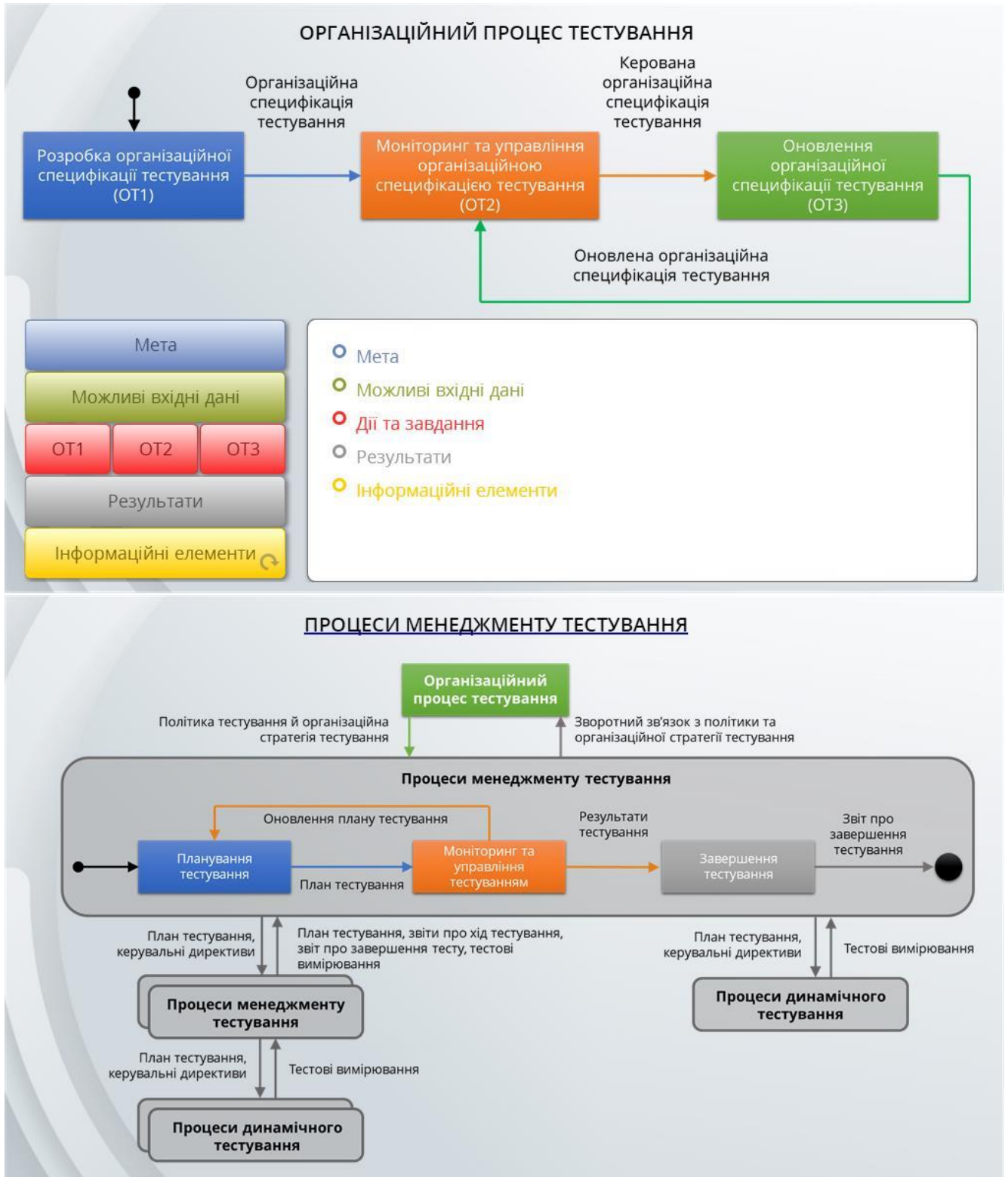
ЛІТЕРАТУРА

1. ISO/IEC/IEEE 29119-2:2013 Software and systems engineering — Software testing — Part 2: Test processes
2. ISO/IEC/IEEE 29119-4:2015 Software and systems engineering — Software testing — Part 4: Test techniques
3. ISO/IEC 33063:2015 Information technology — Process assessment — Process assessment model for software testing

КОНТРОЛЬНІ ПИТАННЯ

1. З чого складається організаційний процес тестування?
2. Які етапи процесу менеджменту тестування?
3. Яка сутність процесів динамічного тестування?
4. Які атрибути описують кожний процес тестування?
5. Які завдання процесу планування тестування?

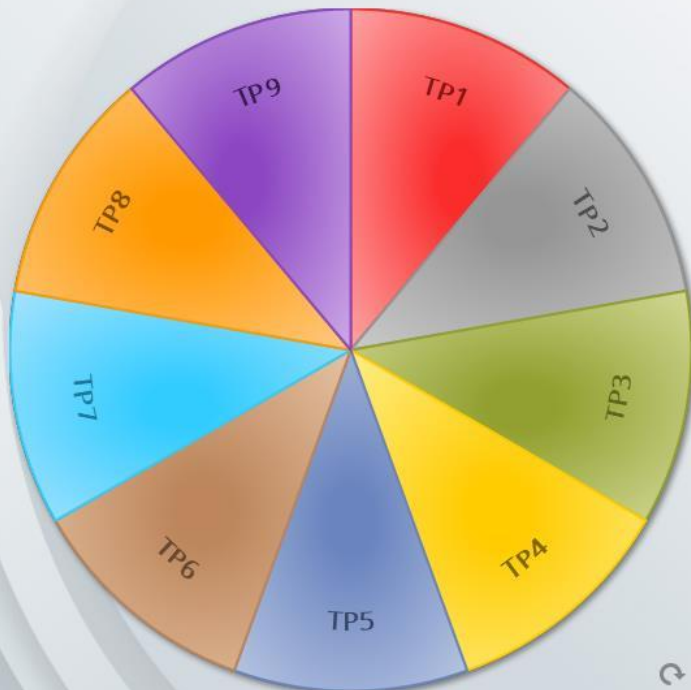
ОПОРНИЙ КОНСПЕКТ



ПРОЦЕС ПЛАНУВАННЯ ТЕСТУВАННЯ



ПРОЦЕС ПЛАНУВАННЯ ТЕСТУВАННЯ



- TP1
- TP2
- TP3
- TP4
- TP5
- TP6
- TP7
- TP8
- TP9

ПРОЦЕС МОНІТОРИНГУ ТА УПРАВЛІННЯ ТЕСТУВАННЯМ



ПРОЦЕС МОНІТОРИНГУ ТА УПРАВЛІННЯ ТЕСТУВАННЯМ



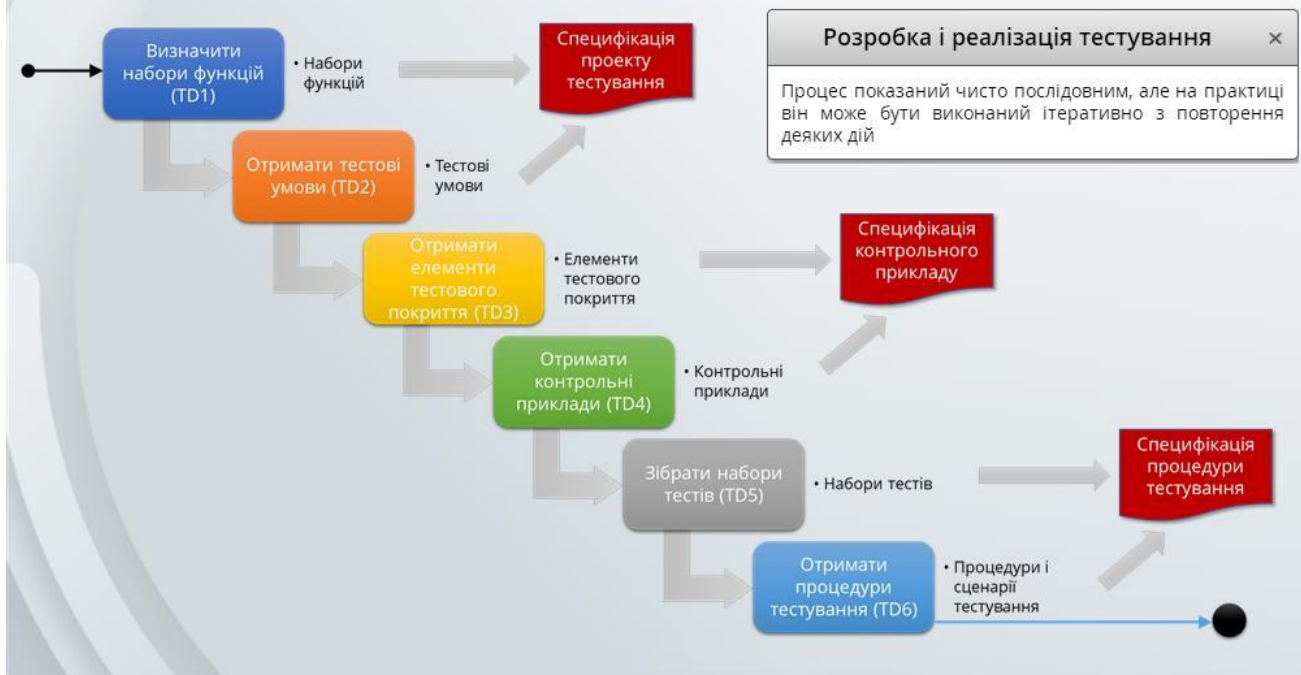
ПРОЦЕС ЗАВЕРШЕННЯ ТЕСТУВАННЯ



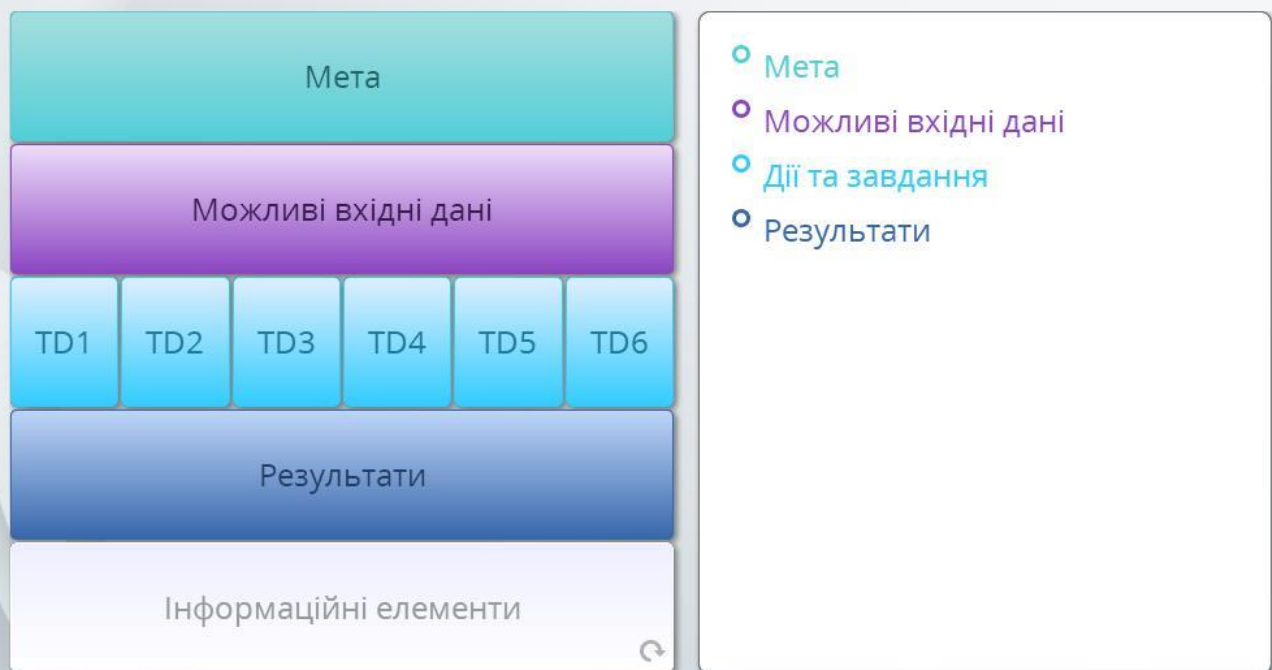
ПРОЦЕСИ ДИНАМІЧНОГО ТЕСТУВАННЯ



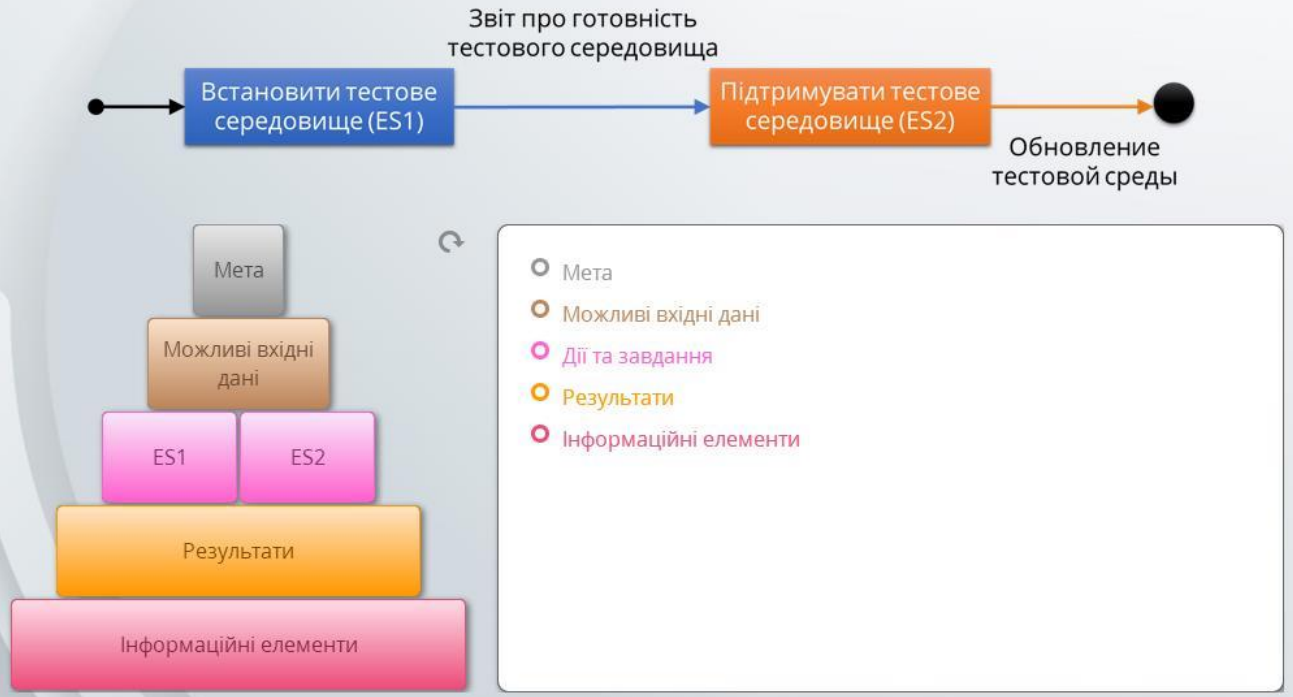
ПРОЦЕС РОЗРОБКИ І РЕАЛІЗАЦІЇ ТЕСТУВАННЯ



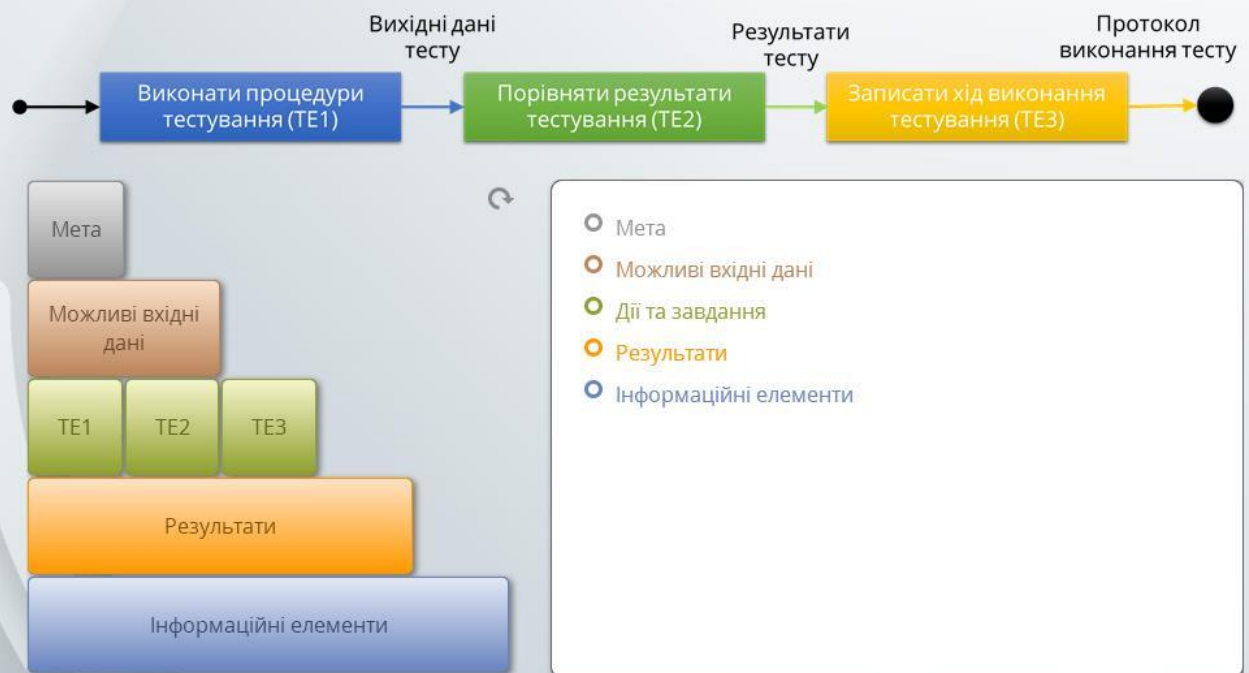
ПРОЦЕС РОЗРОБКИ І РЕАЛІЗАЦІЇ ТЕСТУВАННЯ

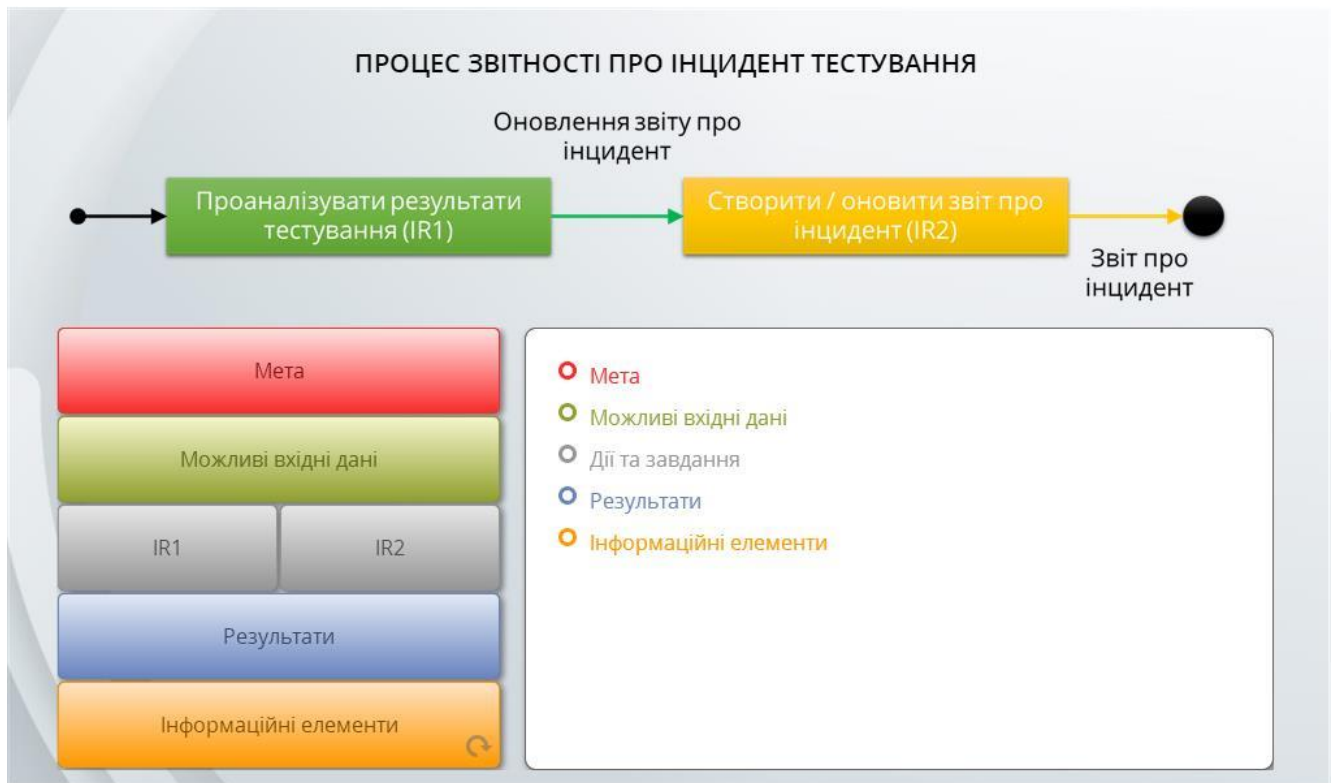


ПРОЦЕС СТВОРЕННЯ ТА ПІДТРИМКИ ТЕСТОВОГО СЕРЕДОВИЩА



ПРОЦЕС ВИКОНАННЯ ТЕСТУ





ЛЕКЦІЯ 16 ТЕСТОВА ДОКУМЕНТАЦІЯ

Мета – знати види та вміст документів, які складають під час тестування програмного забезпечення; вміти обирати та обґрунтовувати необхідний перелік тестової документації для конкретного випадку.

ПЛАН

1. Документація організаційного процесу тестування
2. Документація процесу менеджменту тестування
3. Документація процесу динамічного тестування

ПИТАННЯ ДЛЯ САМОСТІЙНОГО ВИВЧЕННЯ

1. Модель оцінювання процесу для тестування програмного забезпечення

ЛІТЕРАТУРА

1. ISO/IEC/IEEE 29119-3:2013 Software and systems engineering — Software testing — Part 3: Test documentation
2. ISO/IEC 33063:2015 Information technology — Process assessment — Process assessment model for software testing

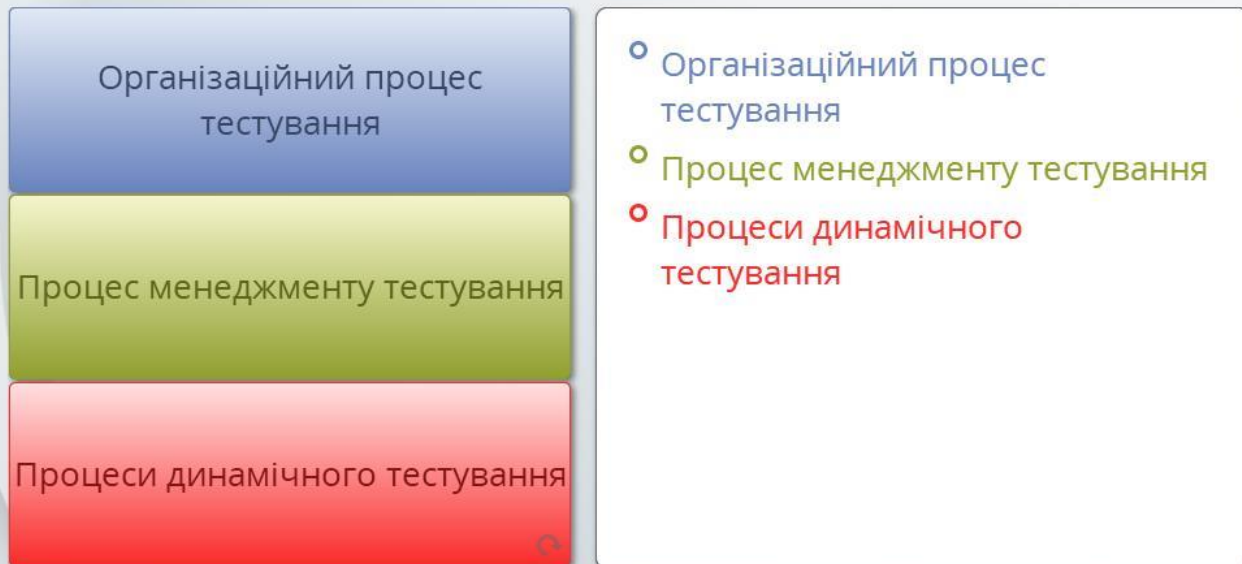
КОНТРОЛЬНІ ПИТАННЯ

1. Яка структура кожного документу?
2. Що пишуть у специфікації та вступі кожного документу?

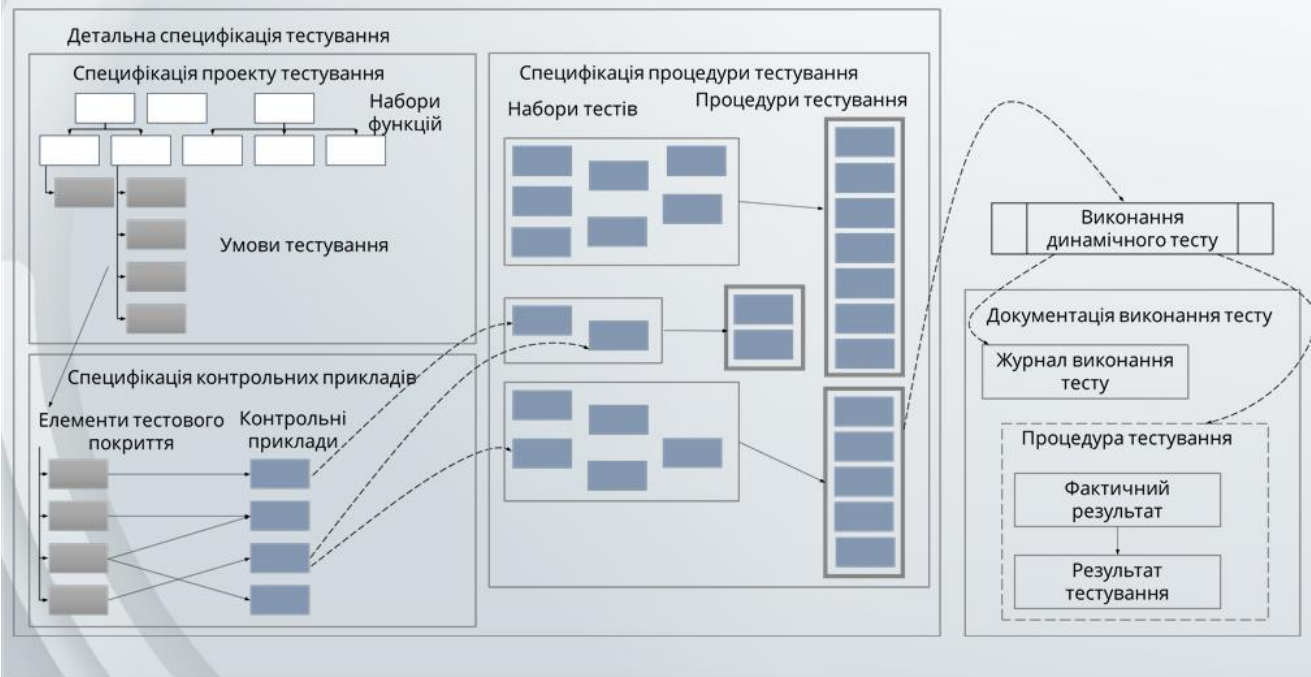
3. Які документи складають під час організаційного процесу тестування?
4. Які документи складають під час менеджменту тестування?
5. Які документи складають під час динамічного тестування?

ОПОРНИЙ КОНСПЕКТ

СТРУКТУРА ТЕСТОВОЇ ДОКУМЕНТАЦІЇ



ІЄРАРХІЯ ДОКУМЕНТАЦІЇ РОЗРОБКИ І РЕАЛІЗАЦІЇ ТЕСТУВАННЯ



СТРУКТУРА КОЖНОГО ДОКУМЕНТА

Специфікація
документа

Вступ

Положення

- ✓ Загальні відомості
- ✓ Унікальна ідентифікація документа
- ✓ Організація, яка оформлює документ
- ✓ Повноваження щодо затвердження
- ✓ Історія змін

Далі

- ✓ Галузь застосування
- ✓ Посилання
- ✓ Глосарій
- ✓ Умовні позначення

Далі

ДОКУМЕНТАЦІЯ ОРГАНІЗАЦІЙНОГО ПРОЦЕСУ ТЕСТУВАННЯ

ПОЛІТИКА

- Цілі тестування
- Процес тестування
- Організаційна структура тестування
- Підготовка тестерів
- Етика тестера
- Стандарти
- Інші відповідні політики
- Оцінка вартості тестування
- Архівація і повторне використання активу тестування
- Удосконалення процесу тестування



- у масштабах проекту
- для конкретних підпроцесів

ДОКУМЕНТАЦІЯ ПРОЦЕСУ МЕНЕДЖМЕНТУ ТЕСТУВАННЯ. ПЛАН ТЕСТУВАННЯ

✓	Контекст тестування
✓	Обмін інформацією про тестування
✓	Реєстр ризиків
✓	Стратегія тестування
✓	Дії та оцінка тестування
✓	Комплектність персоналу
✓	Розклад

Контекст тестування:

1. Проект(и) / підпроцес(и) тестування
2. Елемент(и) тестування
3. Область застосування тестування
4. Припущення і обмеження
5. Зацікавлені сторони

Стратегія тестування:

1. Підпроцеси тестування
2. Практичні результати тестування
3. Методика проектування тестування
4. Критерії завершення тестування
5. Необхідні метрики
6. Вимоги до тестових даних
7. Вимоги до програмного середовища
8. Повторне тестування і регресійне тестування
9. Критерії припинення та поновлення
10. Відхилення від організаційної стратегії тестування

Комплектність персоналу:

1. Ролі, дії та відповідальність
2. Потреба у додатковому персоналі
3. Потреба у навчанні

Реєстр ризиків:

1. Ризики проекту
2. Ризики продукту

ДОКУМЕНТАЦІЯ ПРОЦЕСУ МЕНЕДЖМЕНТУ ТЕСТУВАННЯ

ЗВІТ ПРО ХІД ТЕСТУВАННЯ

- 1 Звітний період
- 2 Прогрес щодо плану тестування
- 3 Фактори, що блокують прогрес
- 4 Показники тестування
- 5 Нові та змінені ризики
- 6 Заплановане тестування

ЗВІТ ПРО ЗАВЕРШЕННЯ ТЕСТУВАННЯ

- 1 Зведення виконаного тестування
- 2 Відхилення від плану тестування
- 3 Оцінка завершення тестування
- 4 Фактори, які перешкоджають
- 5 Показники тестування
- 6 Залишкові ризики
- 7 Практичні результати тестування
- 8 Активи тестування, що допускають повторне використання
- 9 Накопичений досвід

ДОКУМЕНТАЦІЯ ПРОЦЕСІВ ДИНАМІЧНОГО ТЕСТУВАННЯ

Специфікація проекту
тестування

Набори функцій

Тестові умови

- Специфікація проекту тестування
- Набори функцій
- Тестові умови

Специфікація
контрольних прикладів

Елементи тестового
покриття

Контрольні приклади

- Специфікація контрольних прикладів
- Елементи тестового покриття
- Контрольні приклади

ДОКУМЕНТАЦІЯ ПРОЦЕСІВ ДИНАМІЧНОГО ТЕСТУВАННЯ

Специфікація процедур тестування



Специфікація процедур тестування

представляє в порядку виконання контрольні приклади в заданих наборах тестів разом з усіма відповідними діями, які можуть знадобитися для встановлення початкових вихідних умов, і всіма необхідними діями після виконання прикладів

Набори тестів

Процедури тестування

ВИМОГИ ДО ТЕСТОВИХ ДАНИХ

✓	Загальні відомості
✓	Унікальний ідентифікатор
✓	Опис
✓	Відповідальність
✓	Потрібний період
✓	Необхідність скидання
✓	Архівація або утилізація

Далі

ДОКУМЕНТАЦІЯ ПРОЦЕСІВ ДИНАМІЧНОГО ТЕСТУВАННЯ

1

ВИМОГИ ДО
ТЕСТОВОГО
СЕРЕДОВИЩА

Загальні відомості
Опис
Потрібний період

Унікальний ідентифікатор
Відповідальність

2

ЗВІТ ПРО
ГОТОВНІСТЬ
ТЕСТОВИХ
ДАНИХ

Загальні відомості
Опис стану

Унікальний ідентифікатор

3

ЗВІТ ПРО
ГОТОВНІСТЬ
ТЕСТОВОГО
СЕРЕДОВИЩА

Загальні відомості
Опис стану

Унікальний ідентифікатор



Фактичні результати - це запис результату виконання контрольного прикладу в процедурі тестування. Порівняння фактичних результатів з очікуваними визначає результат тестування.

Фактичні результати не завжди формально реєструються. Деякі типи систем (наприклад, в разі регулювання критично важливої захищеності) можуть зажадати повного документування фактичних результатів, а для інших (наприклад, з високою цілісністю даних або вимогами надійності) можна прийняти рішення - записувати фактичні результати повністю. Запис може проводитися автоматизованим інструментом під час виконання тесту.

Деякі контрольні приклади можуть включати дії, які забезпечують результати, які є проміжними, але не є частиною фактичних результатів виконання контрольного прикладу. Вони можуть бути задокументовані або окремо в журналі тестування, або разом з фактичними результатами. В останньому випадку необхідно забезпечити чітку відмінність фактичних результатів від проміжних.

У разі необхідності фактичні результати зазвичай документуються безпосередньо в процедурі тестування в спеціальних полях, для цього зарезервованих. Тому фактичні результати зазвичай не розглядаються як незалежний документ.

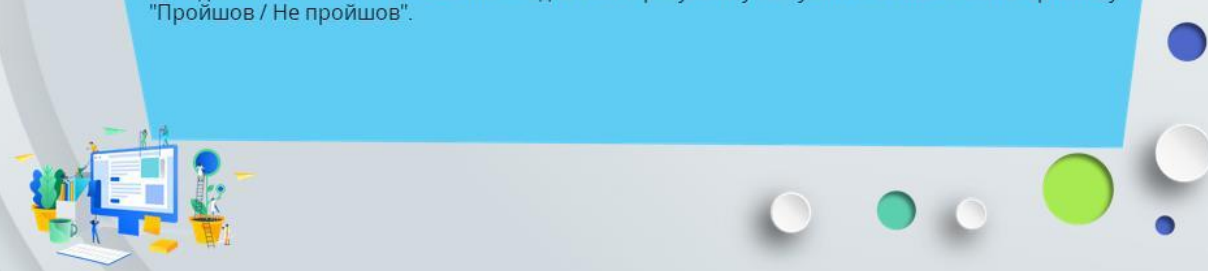




Результат тестування - це запис про те, чи пройшло успішно виконання конкретного контрольного прикладу чи ні, тобто чи відповідають фактичні результати очікуваним або спостерігалися відхилення або заплановане виконання контрольного прикладу неможливо.

Результат тестування для контрольного прикладу зазвичай документується безпосередньо в процедурі тестування в спеціальному зарезервованому для цього полі, тому результат тестування зазвичай не розглядається як незалежний документ.

Іноді цей процес повністю автоматизований за допомогою інструменту, який би порівнював фактичні результати з очікуваними і надає звіт про минулі контрольні приклади, відмови або випадки неможливості виконання. Іноді в якості результату тестування вживають альтернативу "Пройшов / Не пройшов".



ДОКУМЕНТАЦІЯ ПРОЦЕСІВ ДИНАМІЧНОГО ТЕСТУВАННЯ

ЖУРНАЛ ВИКОНАННЯ ТЕСТУ

1	Загальні відомості про подію
2	Унікальний ідентифікатор
3	Дата, час події
4	Опис події
5	Вплив події

ЗВІТ ПРО ІНЦИДЕНТ ТЕСТУВАННЯ

1	Інформація про час	
2	Ініціатор	3 Контекст
4	Опис інциденту	5 Вплив події
6	Оцінка серйозності	7 Оцінка пріоритету
8	Ризик	9 Стан інциденту