

Software  
architecture

**D3S**

# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Луцьк - 2022

*Лектор: д.т.н., проф. І.Є.Андрущак*

# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



## ЛЕКЦІЯ 2. ОСОБЛИВОСТІ ПРОЕКТУВАННЯ АПЗ

### 1. Проектування та його особливості

**Проектування** - процес створення проекту архітектури, прототипу, прообразу майбутнього об'єкта, стану та способів його виготовлення. У проектуванні застосовують системний підхід, який полягає у встановленні структури системи, типу зв'язків, визначенні атрибутів, аналізуванні впливів зовнішнього середовища.

**Проектування** - це комплекс робіт який складається з пошуку, досліджень, розрахунків та розрахування з метою отримання опису достатнього для створення нового об'єкту або виробу, його реконструкції, модернізації, що відповідає заданим вимогам.

У техніці - розробка проектної, конструкторської та іншої технічної документації, призначеної для забезпечення будівництва, створення нових видів та зразків.

В процесі проектування виконуються технічні та економічні розрахунки, схеми, графіки, пояснювальні записки, кошториси, калькуляції та описи.



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## ТЕХНІЧНЕ ЗАВДАННЯ (ТЗ)

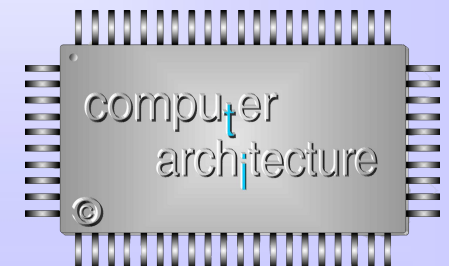
**Проект** (від лат. Projectus - кинутий вперед) - це обмежена в часі, ресурсах та вимогах якості унікальна сукупність процесів, направлена на створення нової архітектури.

**В залежності від сфери здійснення проекту виділяють кілька типів проектів:**

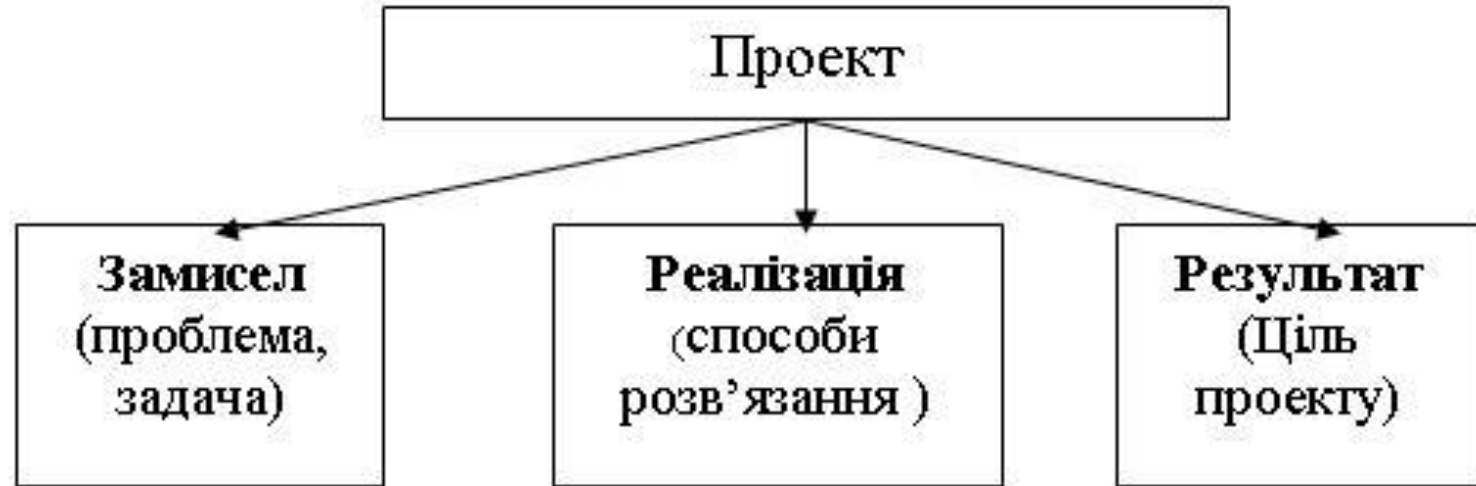
- *технічний проект* - направлений на створення нового продукту;
- *організаційний проект* - направлений на створення або реорганізацію організації;
- *соціальний проект* - направлений на отримання певного соціального ефекту.

**В залежності від масштабу виділяють наступні типи проектів:**

- малі проекти, мають бюджет до 0.5 млн євро;
- середні проекти, мають бюджет до 0.5 млрд євро;
- мегапроекти, мають бюджет більше 0.5 млрд євро.



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



**Проекти також поділяються залежно до часу реалізації:**

- короткострокові;
- довгострокові).

**За країнами учасників проекту:**

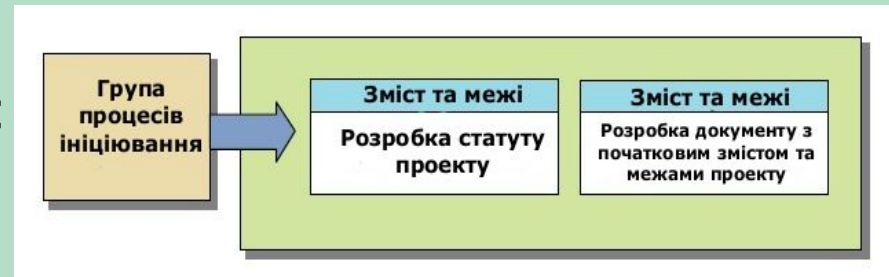
- національні;
- міжнародні.

**Проекти можуть бути об'єднані в:**

- програму проектів для досягнення єдиного результату;
- портфель проектів організації для ефективнішого управління, портфель проектів може включати також програми.

Досягнення мети проекту вимагає отримання результатів, що відповідають певним заздалегідь вимогам, у тому числі обмеження на отримання результатів, таких як час, гроші і ресурси.

Вигоди від реалізації проекту - грошове вираження сукупної вартості всіх товарів, послуг і інших вигод, що виникають як результат капіталовкладень у даний проект.



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

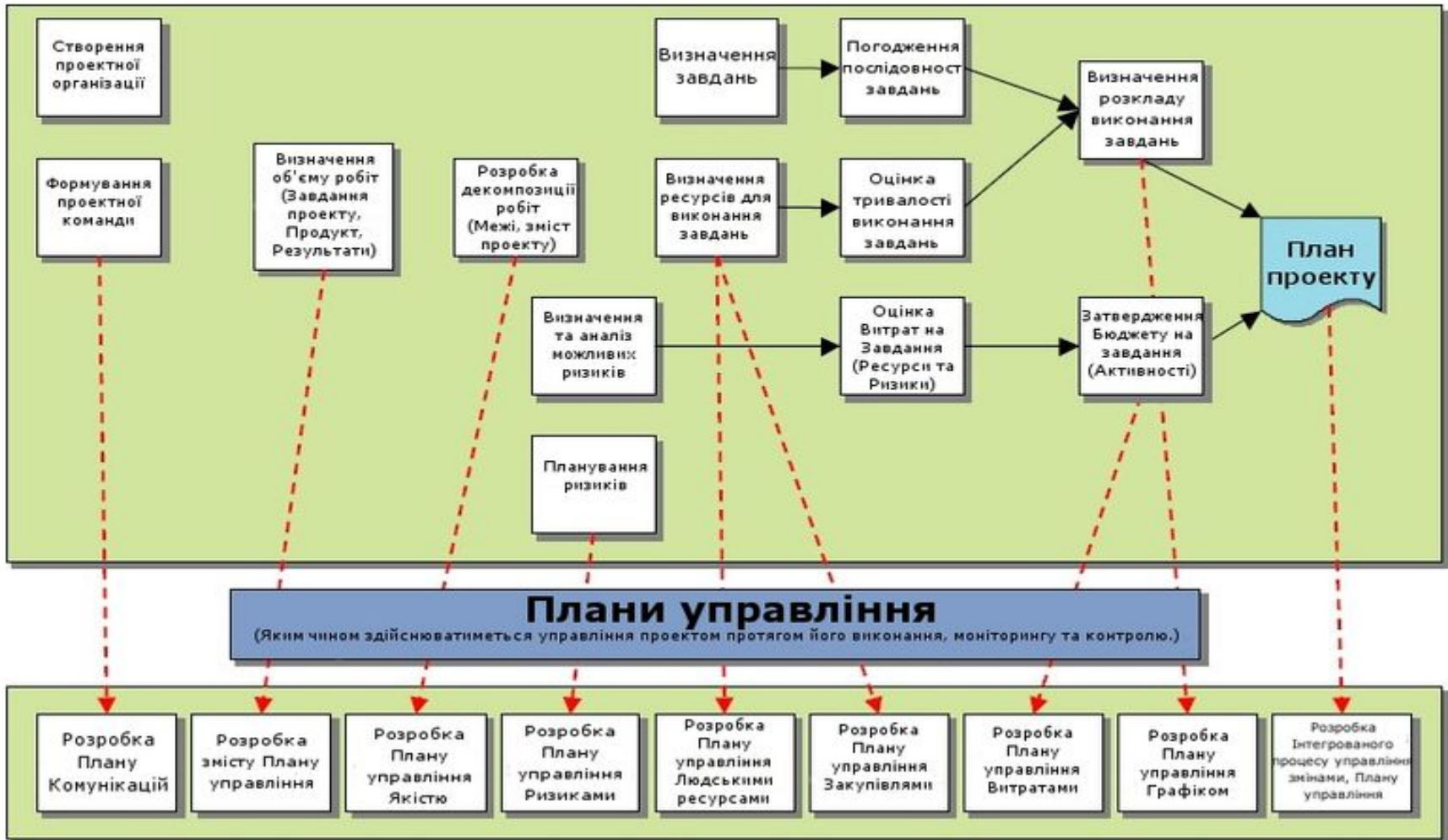
D3S



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



## Група процесів планування



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## Стадії проектування (цикл розробки ПЗ)

**1. Аналіз вимог** - починається із стадії аналізу, під час якого учасники процесу обговорюють вимоги, що висуваються до кінцевого продукту.

**Мета цієї стадії** – визначення детальних вимог до системи. Окрім того, необхідно впевнитися в тому, що всі учасники вірно зрозуміли поставлені задачі і те, як саме кожна вимога буде реалізована на практиці.

Таким чином, цей етап передбачає **збір вимог** до програмного забезпечення, що розробляється, їх **систематизацію, документування, аналіз, а також виявлення ти вирішення протиріч.**





# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



## **2. Проектування.**

На стадії проектування (яка також називається стадією дизайну та архітектури) програмісти та системні архітектори, керуючись вимогами, розробляють високорівневий дизайн системи.

Різноманітні технічні питання, які виникають в процесі проектування, обговорюються зі всіма зацікавленими сторонами, включаючи замовника. Визначаються технології, які будуть використовуватися в проекті, завантаженість команди, обмеження, часові рамки та бюджет. Відповідно з уточненими вимогами обираються максимально відповідні проектні рішення.

Затверджений дизайн системи визначає перелік програмних компонентів, які необхідно розробити, взаємодію з третіми сторонами, функціональні характеристики програми, бази даних, які необхідно використовувати та багато іншого.

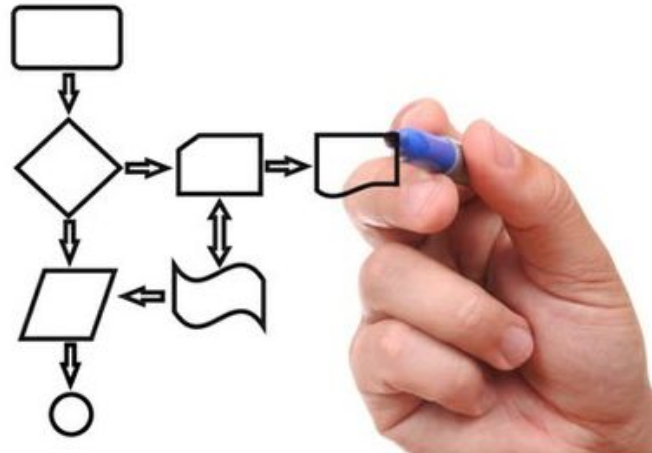
# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Дизайн, як правило, фіксується окремим документом – **дизайн-специфікацією** (Design Specification Document, DSD).

На цьому етапі спрощення процесу візуалізації використовуються як так звані нотації – схематичні зображення системи, що розробляється. Основі нотації:

- [Блок-схеми](#);
- [ER-діаграми](#);
- [UML-діаграми](#);
- Макети – наприклад, намальований в фотошопі прототип сайту.







# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



## 4. Документація

Цей етап виділяють достатньо умовно, оскільки, як ми бачили, ті чи інші документи створюються на всіх стадіях життєвого циклу програми.

Всього існує **чотири рівня документації**:

- **Архітектурна (проектна)** - документи, що описують моделі, методології, інструменти та засоби розробки, обрані для даного проекту.
- **Технічна** – вся документація, що супроводжує розробку. Сюди входять різноманітні документи, що пояснюють розробку системи на рівні окремих модулів (HTML-документи).



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



– **Користувацька** – включає довідникові та пояснюючі матеріали, необхідні кінцевому користувачу для роботи з системою (Readme та Userguide, розділ довідки по програмі).

– **Маркетингова** – включає рекламні матеріали, які супроводжують випуск продукту. Її мета – в яскравій формі презентувати функціональність та конкурентні переваги продукту.



| **Documentation**

# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## 5. Тестування.

Тестувальники займаються пошуком дефектів у програмному забезпеченні і порівнюють описану у вимогах поведінку системи з реальною. У фазі тестування виявляються пропущені при розробці баги. При виявленні дефекту, тестувальник складає звіт про помилку, який передається розробникам. Останні його виправляють, після чого тестування повторюється – але цього разу для того, щоб переконатися, що проблема була виправлена, і саме виправлення не стало причиною появи нових дефектів в продукті.

Тестування повторюється до тих пір, поки не будуть досягнуті критерії його закінчення.



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



## ***6. Впровадження та супровід***

Коли програма протестована і в ній більше не залишилося серйозних дефектів, приходить час релізу і передачі її кінцевим користувачам. Після випуску нової версії програми в роботу включається відділ технічної підтримки. Його співробітники забезпечують зворотній зв'язок з користувачами, їх консультування та підтримку. Крім того, команда технічної підтримки допомагає збирати та систематизувати різні метрики – показники роботи програми в реальних умовах.





# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## ТЕХНІЧНЕ ЗАВДАННЯ (ТЗ)

- (англ. *statement of work; SOW*) - документ, що встановлює основне призначення, показники якості, техніко-економічні та спеціальні вимоги до виробу, обсягу, стадії розроблення та складу конструкторської документації.

**При розробці технічного завдання** використовуються наступні інформаційні матеріали:

- науково-технічна інформація;
- патентна інформація;
- характеристика ринку збуту;
- характеристика виробництва (технічна оснащеність, кваліфікація кадрів, технологічна дисципліна), на якому виріб планується виготовляти.



При розробці технічного завдання на вироби підвищеної складності попередньо проробляється так званий **аванпроект**, який дозволяє попередньо поглиблено пропрацювати комплекс питань, що визначають необхідність та доцільність створення такого виробу. Розробка аванпроекту повинна гарантувати можливість створення продукції, що відповідає за техніко-економічними показниками світовому рівню.

# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Згідно з чинними стандартами ТЗ повинно включати в себе такі відомості про об'єкт розробки:

## ***1. Найменування об'єкта розробки, та область застосування.***

- повне найменування об'єкта та умовне позначення;
- шифр теми або шифр (номер) договору;
- перелік документів, на підставі яких створюється проект, ким і коли затверджені ці документи;
- планові терміни початку та закінчення робіт із створення об'єкта.

## **2. Підстава для розробки та назва проектної організації:**



- найменування підприємств розробника і замовника системи та їхні реквізити;
- перелік юридичних та фінансових документів, на підставі яких створюється система, ким і коли затверджені ці документи;
- відомості про джерела та порядок фінансування робіт.





# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

**3. Мета розробки.**

**4. Джерела розробки.** Тут повинні бути перераховані документи та інформаційні матеріали (техніко-економічне обґрунтування, інформаційні посилання на вітчизняні і зарубіжні аналоги та ін.), на підставі яких розроблялося ТЗ і які мають бути використані при створенні системи.

**5. Технічні вимоги, які включають:**

Computer Architecture

- склад об'єкта та вимоги до його конструктивного виконання;
- показники призначення та економічного використання матеріалів;
- вимоги до надійності;
- моги безпеки при роботі обладнання;
- вимоги до складових частин продукції і експлуатаційних матеріалів;
- вимоги патентної чистоти;
- вимоги експлуатації, вимоги до технічного обслуговування і ремонту;
- вимоги до категорії якості.



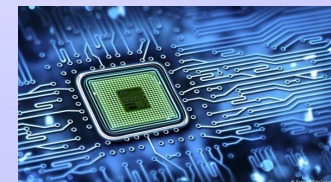
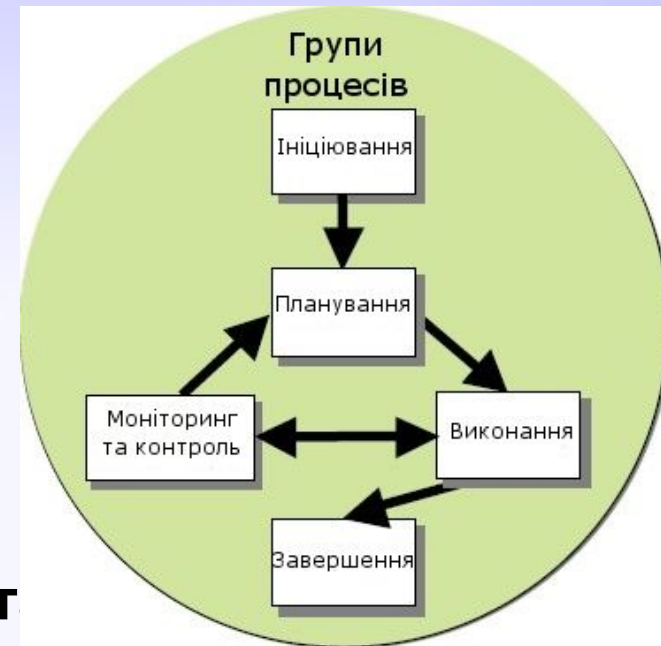
# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## 6. Економічні показники:

- гранична ціна;
- економічний ефект;
- термін окупності витрат на розробку і освоєння об'єкта;
- допустима річна потреба в об'єкті проектування .

## 7. Порядок контролю і приймання об'єкт

- види, склад, обсяг і методи випробувань системи та її складових частин (види випробувань відповідно до чинних норм, які поширюються на систему, що розробляється);
- загальні вимоги до приймання робіт (продукції) за стадіями (перелік учасників, місце і терміни проведення), порядок узгодження і затвердження приймальної документації;
- статус приймальної комісії.



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



## ЕКСКІЗНИЙ ПРОЕКТ

**Ескізний проект** (англ. *Preliminary design*) - проектна конструкторська документація, яка містить принципові конструктивні рішення і дає загальне уявлення про будову та принцип дії виробу, а також дані, що визначають його відповідність призначенню.

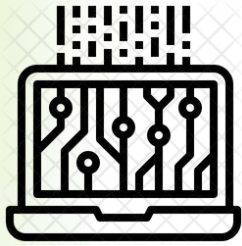
Ескізний проект розробляють з метою встановлення принципів (конструктивних, схемних) рішень виробу, що дають загальне уявлення про принцип роботи або будову виробу, коли це доцільно зробити до розробки технічного проекту чи робочої документації.

На стадії розробки ескізного проекту розглядають варіанти виробу і (або) його складових частин. Ескізний проект може розроблятися і без розгляду на цій стадії різних варіантів.

У комплект документів ескізного проекту включають конструкторські документи, згідно з ДСТУ передбачені технічним завданням і протоколом розгляду технічної пропозиції.



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ





# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

При розробці ескізного проекту виконують роботи, необхідні для реалізації поставлених до виробу вимог і дозволяють встановити принципові рішення. Перелік необхідних робіт визначається розробником в залежності від характеру і призначення виробу. У загальному випадку ***при розробці ескізного проекту проводять наступні роботи:***

- виконання варіантів можливих рішень,;
- виготовлення і випробування макетів і (або) розробка та аналіз електронних макетів з метою перевірки принципів роботи виробу його складових частин;
- розробку та обґрунтування технічних рішень, спрямованих на забезпечення показників надійності, установлених технічним завданням та технічною пропозицією;
- оцінку виробу;

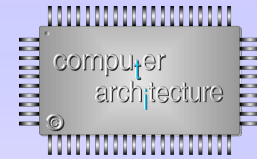
IEEE

COMPUTER  
ARCHITECTURE

LETTERS



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



- перевірку варіантів на патентну частоту і конкурентоспроможність, оформлення заявок на винаходи;
- перевірку відповідності варіантів вимогам техніки безпеки;
- порівняльну оцінку розглянутих варіантів;
- вибір оптимального варіанту (варіантів) виробу, обґрунтування вибору, прийняття принципів рішень;
- підтвердження пропонованих до виробу вимог (технічних характеристик, показників якості тощо), установлених технічним завданням і технічною пропозицією, і визначення техніко-економічних характеристик і показників, не встановлених технічним завданням і технічною пропозицією;
- виявлення на основі прийнятих принципів рішень нових виробів і матеріалів, які повинні бути розроблені іншими підприємствами (організаціями), розроблення технічних вимог до цих виробів і матеріалів.



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТЕХНІЧНИЙ ПРОЕКТ



**Технічний проект** - стадія розробки виробу і проектна конструкторська документація, яка містить остаточне технічне рішення і дає повне уявлення про будову розроблюваного виробу або стадія створення автоматизованої системи.

Технічний проект розробляється на підставі затвердженого завдання на проектування та техніко-економічного обґрунтування.

**Основні етапи виконання робіт при розробці технічного проекту:**

- розробка конструкторської документації технічного проекту з присвоєнням документам літери «Т».
- виготовлення та випробування макетів (за необхідності).
- розгляд та затвердження технічного проекту.



Analyze  
Position

Prioritize  
Criteria

STRATEGIC  
PLANNING

Review

Define  
Goals

Define  
Accountability



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



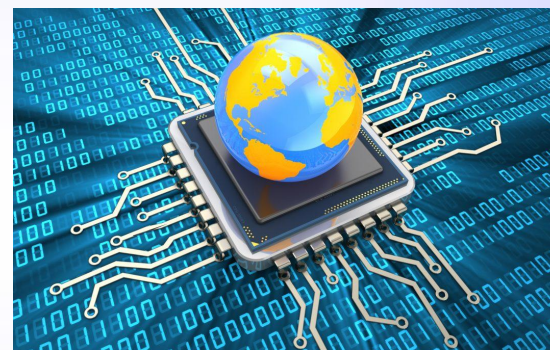
## 2. Життєвий цикл АПЗ

Проект може бути розбитий як на підпроекти, так і на фази.

Сукупність фаз являє собою життєвий цикл проекту, який має 5

фаз:

- ініціація (англ. *Initiating*);
- планування (англ. *Planning*);
- виконання (англ. *Executing*);
- контроль і моніторинг (англ. *Controlling and Monitoring*);
- завершення (англ. *Closing*).



**Життєвий цикл АПЗ** - період часу, який починається з моменту прийняття рішення про необхідність створення певної архітектури і закінчується в момент її повного вилучення з експлуатації

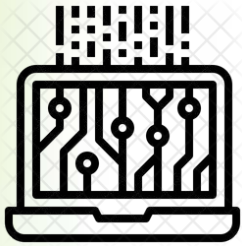
# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

**Життєвий цикл програмного забезпечення** (також має назву цикл розробки) – це умовна схема, що включає в себе окремі етапи, які є стадіями розвитку процесу створення ПЗ. При цьому, на кожному етапі виконуються різні дії.

**Мета використання моделі життєвого циклу** – створити ефективний, економічно вигідний та якісний програмний продукт.



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



**Стадія** - частина процесу створення АПЗ, обмежена певними часовими рамками і закінчується випуском конкретного продукту, що визначається заданими для даної стадії вимогами.

Виділяють наступні стадії:

**1) Початкова стадія** - встановлюється область застосування системи і визначаються граничні умови. На початковій стадії ідентифікуються всі функціональні можливості системи і проводиться опис найбільш істотних з них.

**2) Стадія уточнення** - проводиться аналіз прикладної області, розробляється архітектурна основа інформаційної системи.

**3) Стадія конструювання** - розробляється закінчений виріб, готове до передачі користувачеві. По закінченні цієї стадії визначається працездатність розробленого програмного забезпечення.

**4) Стадія передачі в експлуатацію.**



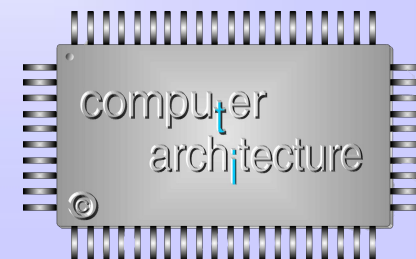


# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

В даний час відомі і використовуються наступні моделі життєвого циклу:

**Каскадна модель (водоспад)** - передбачає послідовне виконання всіх етапів проекту в чітко фіксованому порядку. Перехід на наступний етап означає повне завершення робіт на попередньому етапі.

При моделюванні за принципом «водоспаду» робота над проектом рухається лінійно і послідовно. Перевагою такого підходу є простота його реалізації, недоліком — накопичення можливих на ранніх етапах помилок до моменту закінчення проекту і, як наслідок, зростання ризику провалу проекту, збільшення вартості проекту.



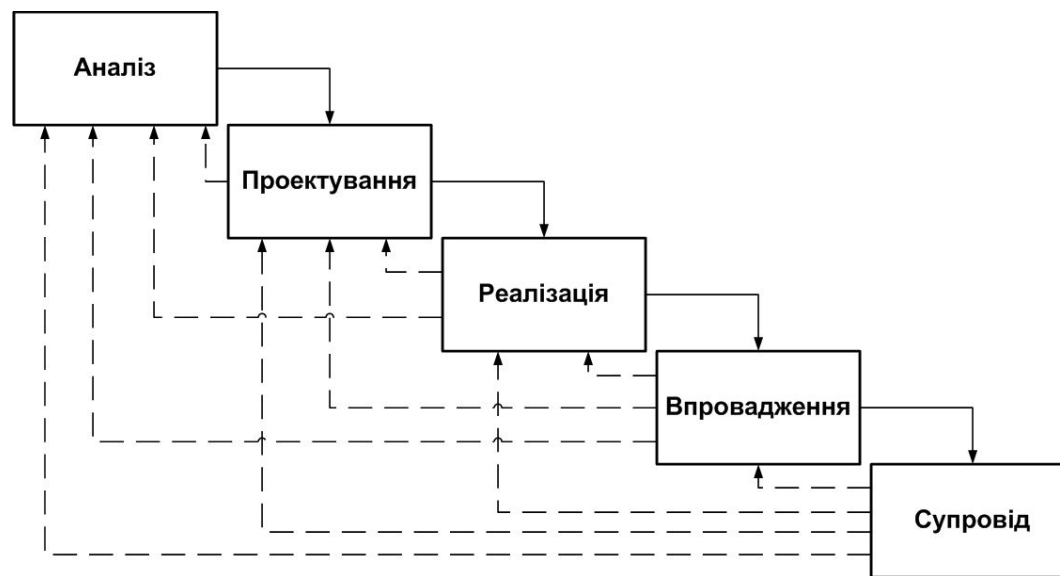
# iteration

## АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Поетапна модель з проміжним контролем - розробка ІС ведеться ітераціями з циклами зворотного зв'язку між етапами.

*Ітеративний підхід* (англ. *Iteration* - повторення) - виконання робіт паралельно з безперервним аналізом отриманих результатів і коригуванням попередніх етапів роботи.

Проект при цьому підході в кожній фазі розвитку проходить повторюваний цикл: Планування - Виконання - Контроль і оцінка (англ. *plan-do-check-act cycle*).

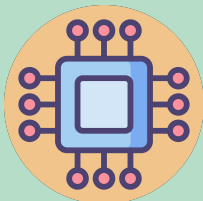




# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## V-подібна модель (V-model)

Суть цієї моделі полягає в тому, що процеси на всіх етапах контролюються, щоб переконатися в можливості переходу на наступний рівень. Вже на стадії написання вимог починається процес тестування.

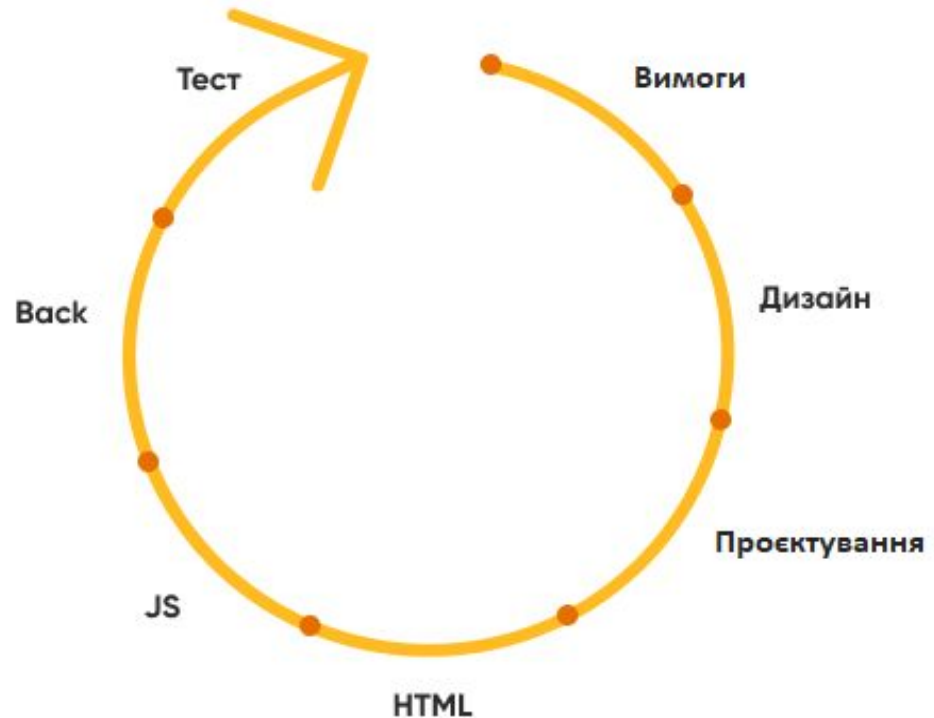


# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



ITERATION

## Ітераційна модель



Даний підхід дозволяє боротися з невизначеністю, знімаючи її етап за етапом, і перевіряти правильність технічного, маркетингового або будь-якого іншого рішення на ранніх стадіях.

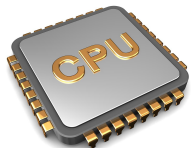
# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



## *Переваги ітеративного підходу:*

- зниження впливу серйозних ризиків на ранніх стадіях проекту, що веде до мінімізації витрат на їх усунення;
- організація ефективного зворотного зв'язку проектною командою зі споживачем, що збільшує вірогідність створення продукту, який реально відповідає їх потребам;
- акцент зусиль на найбільш важливі та критичні напрямки проекту;
- безперервне ітеративне тестування, що дозволяє оцінити успішність всього проекту в цілому;
- раннє виявлення конфліктів між вимогами, моделями і реалізацією проекту;
- більш рівномірне завантаження учасників проекту;
- ефективне використання накопиченого досвіду;
- реальна оцінка поточного стану проекту і, як наслідок, більша впевненість замовників і безпосередніх учасників в його успішному завершенні.

eCAADe



# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



**Спіральна модель** - на кожному витку спіралі виконується створення чергової версії продукту, уточнюються вимоги проекту, визначається його якість, і плануються роботи наступного витка.

Особливу увагу приділяється початковим етапам розробки - аналізу та проектуванню, де реалізація тих чи інших технічних рішень перевіряється і обгрунтовується за допомогою створення прототипів (макетування).



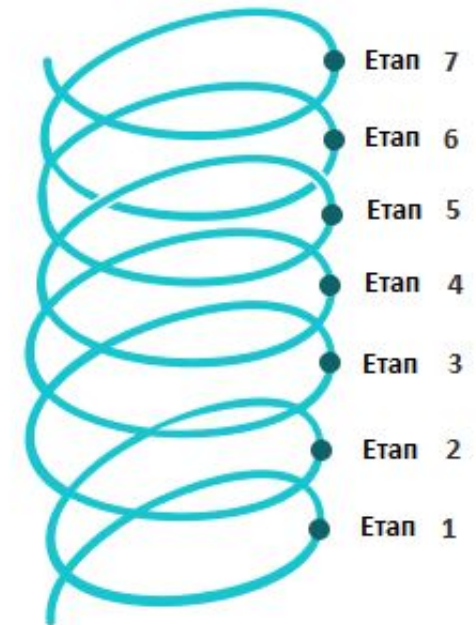
# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



## Спіральна модель



Як працює з середини



Як бачить клієнт

Усі етапи життєвого циклу при спіральній моделі йдуть витками, на кожному з яких відбуваються проектування, кодування, дизайн, тестування і т. д.

# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

В спіральній моделі розглядається залежність ефективності проекту від його вартості з часом. На кожному витку спіралі виконується створення чергової версії продукту, уточнюються вимоги проекту, визначається його якість і плануються роботи наступного витка.

Модель застосовують в сферах, що дозволяють постійне оновлення продукту проекту, наприклад — при розробці програмного забезпечення.



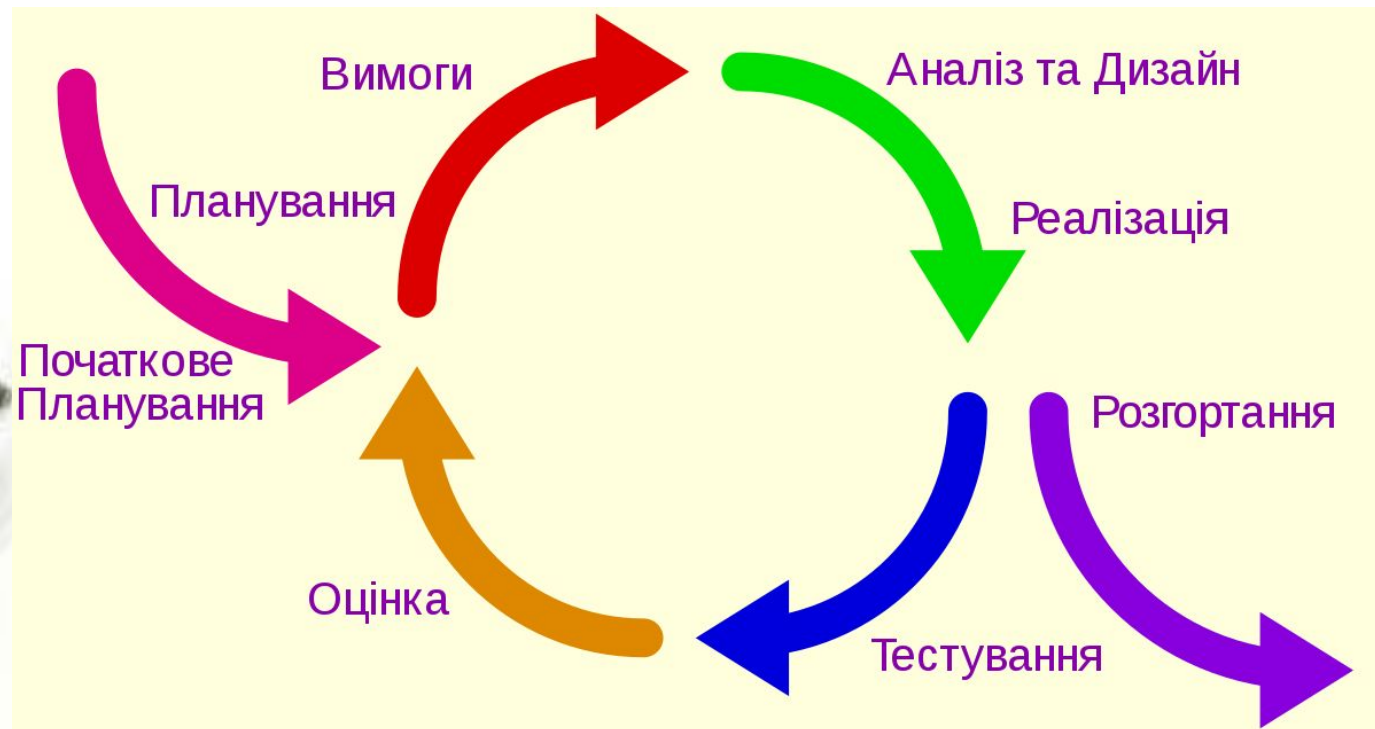
# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



## Інкрементна модель

Принцип, що лежить в основі інкрементної моделі, має на увазі розширення можливостей, добудовування модулів і функцій програми. Буквальний переклад слова інкремент: «збільшення на один».

Це «збільшення на один» застосовується в тому числі для позначення версій продукту.

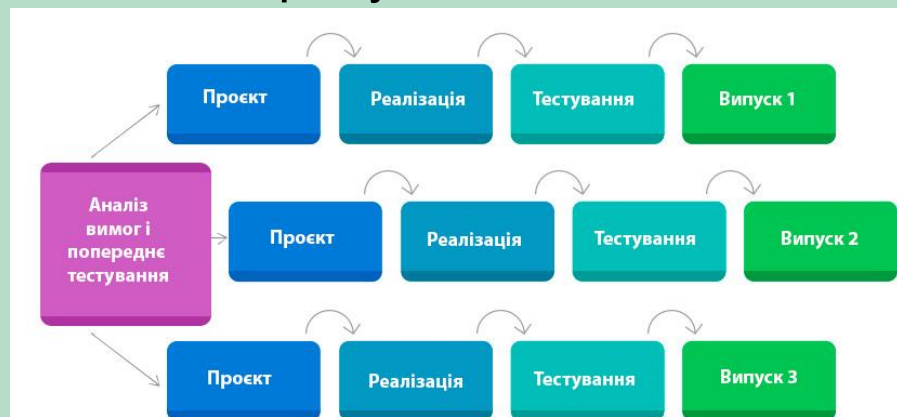


# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## Інкрементна модель (Incremental model)

Відповідно до інкрементної моделі (англ. Increment - збільшення, прирощення) програмне забезпечення розробляється з лінійною послідовністю стадій, але в кілька інкрементів (версій). Таким чином поліпшення продукту проходить заплановано весь час, поки життєвий цикл розробки ПЗ не завершиться.

Вимоги до системи визначаються на самому початку роботи, після чого процес розробки проводиться у вигляді послідовності версій, кожна з яких є закінченим і працездатним продуктом.



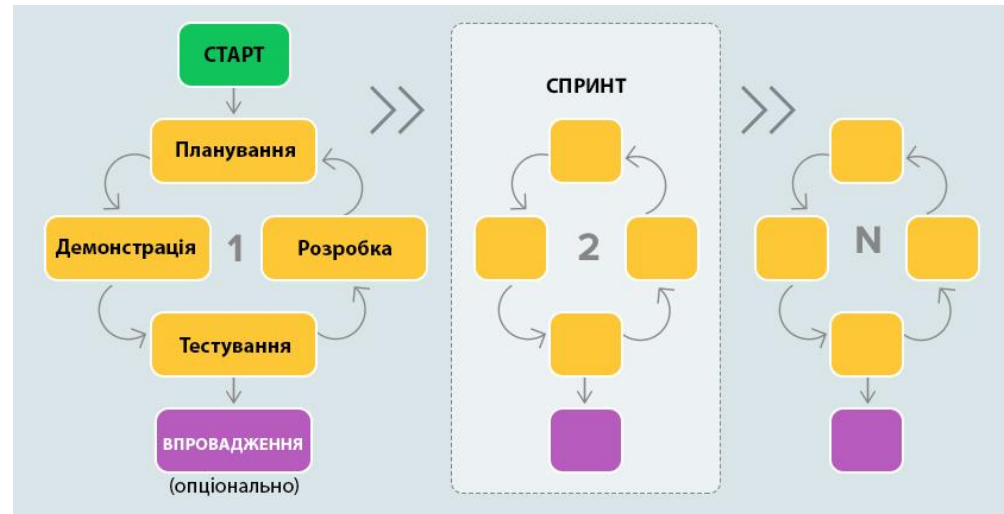


# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



## Гнучка модель (Agile model)

Являє собою сукупність різних підходів до розробки ПЗ. Включає серії підходів до розробки програмного забезпечення, орієнтованого на використання ітеративної розробки (в Scrum ітерації називаються спринтами), динамічне формування вимог і забезпечення їхньої реалізації в результаті постійної взаємодії всередині самоорганізованих робочих груп, що складаються з фахівців різного профілю. Однією з основних ідей Agile є взаємодія всередині команди і з замовником напряду.

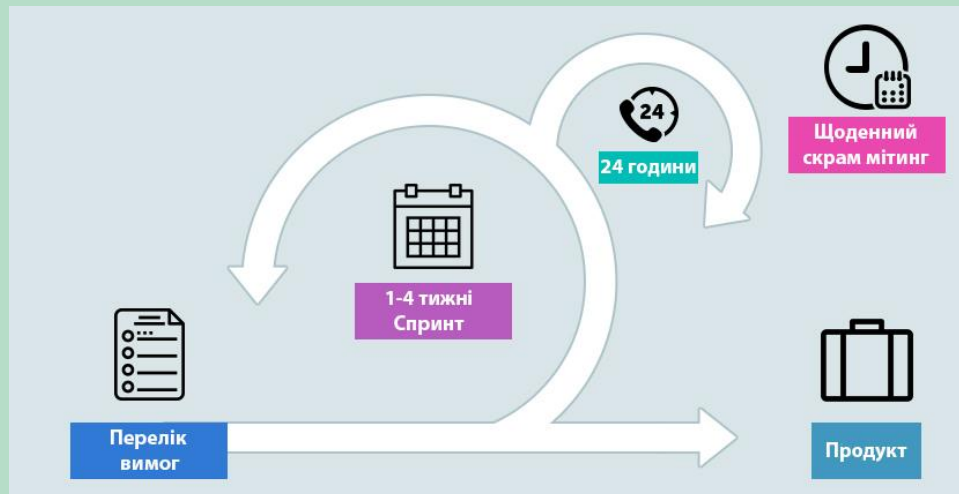


# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

## Скрам (Scrum)

Скрам - це гнучка модель розробки ПЗ, в якій робиться акцент на якісному контролі процесу розробки.

Ролі в методології (Scrum Master, Product Owner, Team) дозволяють чітко розподілити обов'язки в процесі розробки. За успіх Scrum в проєкті відповідає Scrum Master і є сполучною ланкою між менеджментом і командою. За розробку продукту відповідає Product Owner, який також ставить завдання і приймає остаточні рішення для команди.





# АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



**ДЯКУЮ  
ЗА УВАГУ!!!**

