

Технологія створення програмних продуктів

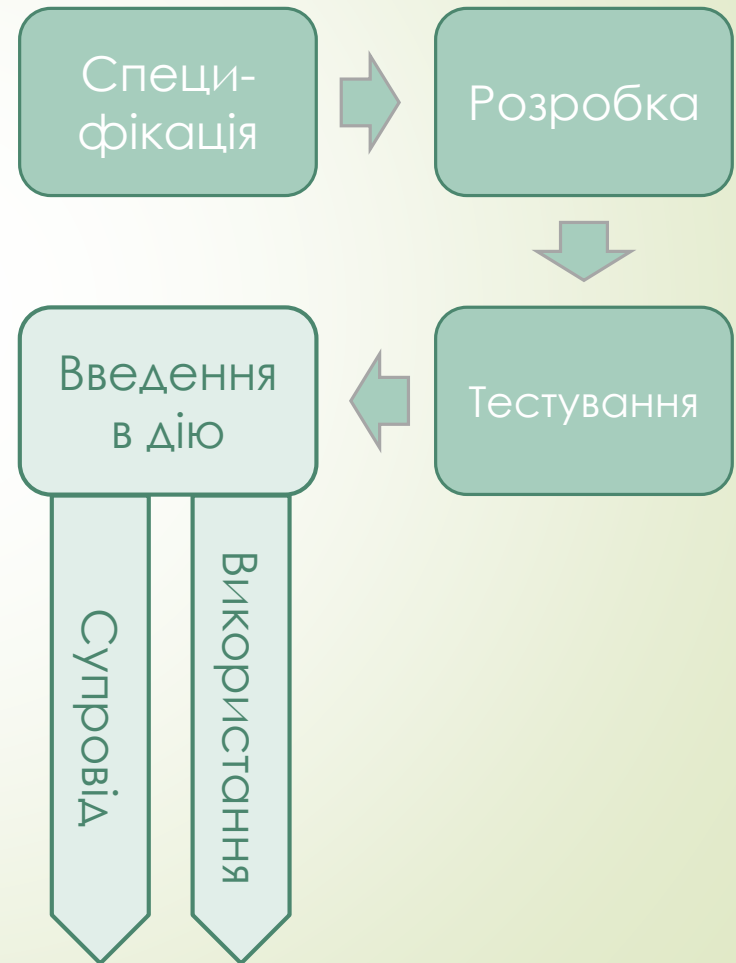
Моделі життєвих
циклів програмних
продуктів



Життєвий цикл програмного забезпечення

Життєвий цикл програмного забезпечення – період часу, що починається з моменту прийняття рішення про необхідність створення програмного продукту і закінчується в момент його повного вилучення з експлуатації.

Цей цикл – процес побудови і розвитку ПЗ.



Основні процеси життєвого циклу програмних систем



Допоміжні процеси життєвого циклу програмних систем



Моделі життєвого циклу програмних продуктів



- Каскадна модель
- V-подібна модель
- Ітеративна модель
- Інкрементна модель
- Спіральна модель
- Еволюційна модель

Каскадна модель



Каскадна модель

○ Недоліки цієї моделі:

- процес створення ПС не завжди укладається в таку жорстку форму і послідовність дій;
- не враховуються потреби користувачів, зміни в зовнішньому середовищі;
- великий розрив між часом внесення помилки і часом її виявлення.

○ Проблеми, які можуть виникнути при використанні даної моделі:

- вимоги недостатньо добре подані;
- система дуже велика за об'ємом, щоб бути реалізованою в цілому;
- швидкі зміни в технології і у вимогах;
- обмежені ресурси (людські, програмні тощо);
- отриманий продукт може виявитися непридатним для використання через неправильне розуміння вимог або функцій системи, а також недостатнього тестування.

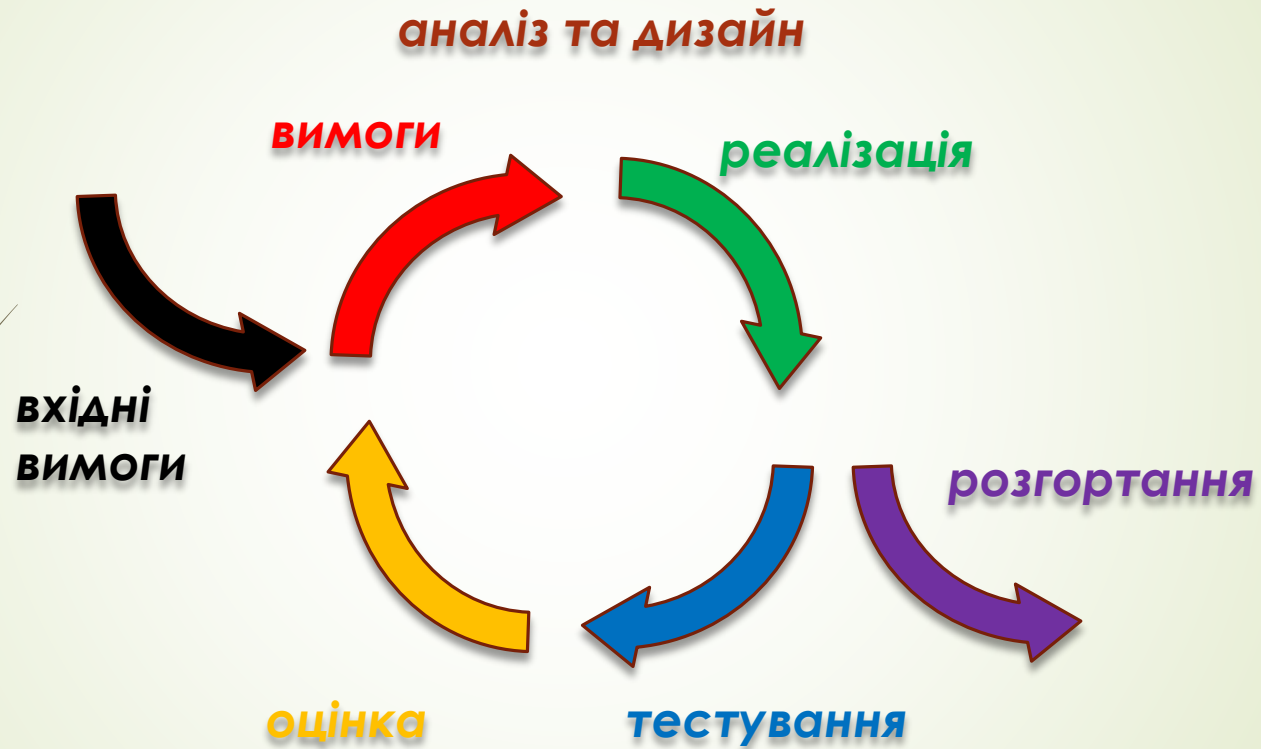
○ Переваги реалізації системи за допомогою каскадної моделі:

- всі завдання підсистем і системи реалізуються одночасно (жодне завдання не забуте), що сприяє встановленню стабільних зв'язків і відносин між ними;
- повністю розроблену систему з документацією на неї легше супроводжувати, тестувати, фіксувати помилки та вносити зміни не безладно, а цілеспрямовано, починаючи з вимог і повторити процес.

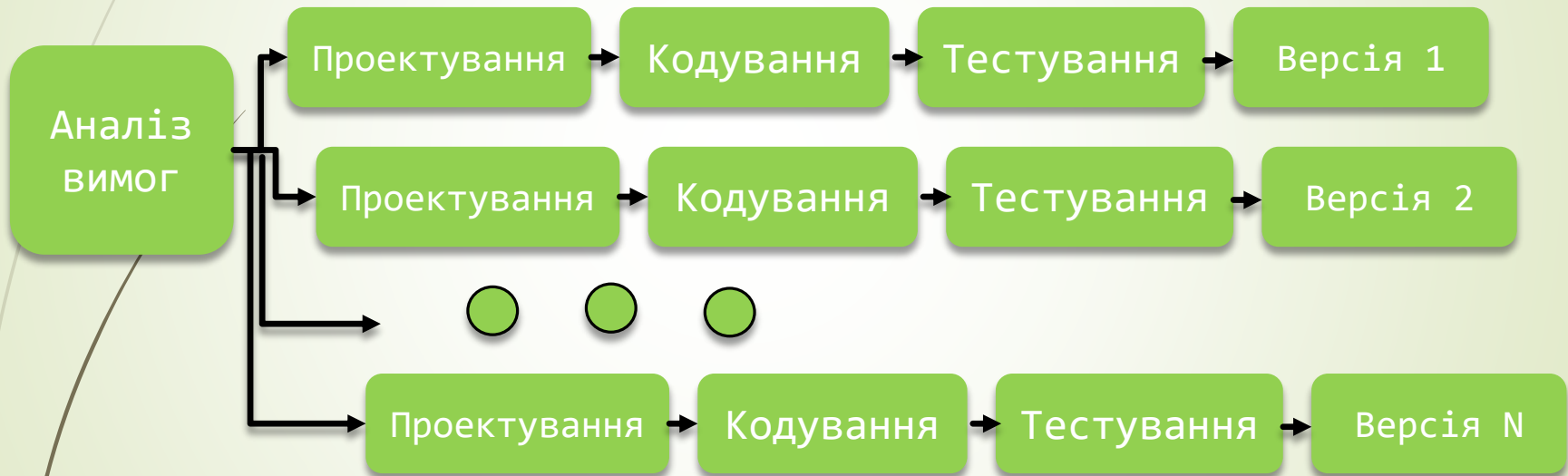
Класична ітеративна модель



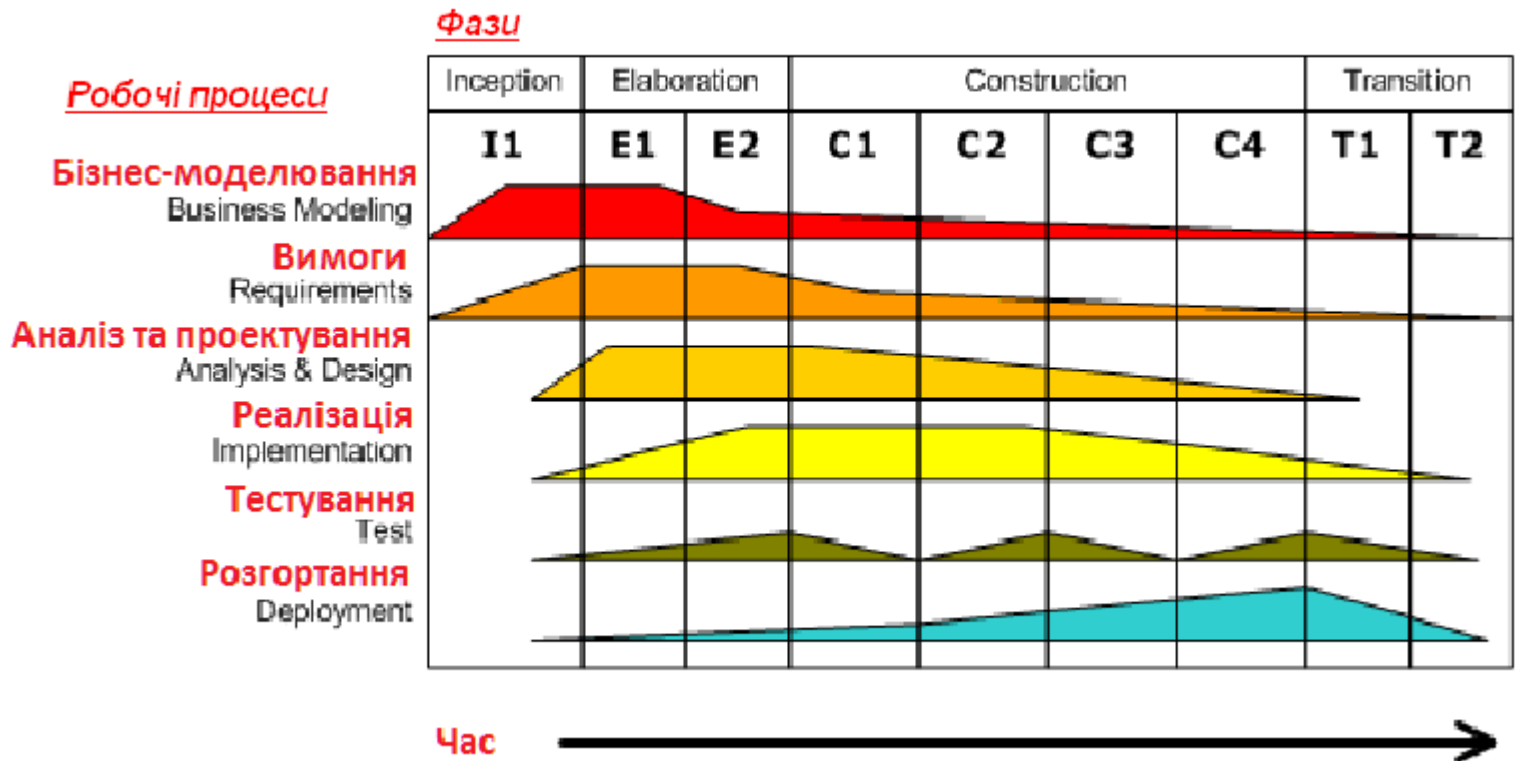
Ітеративна модель



Класична інкрементна модель



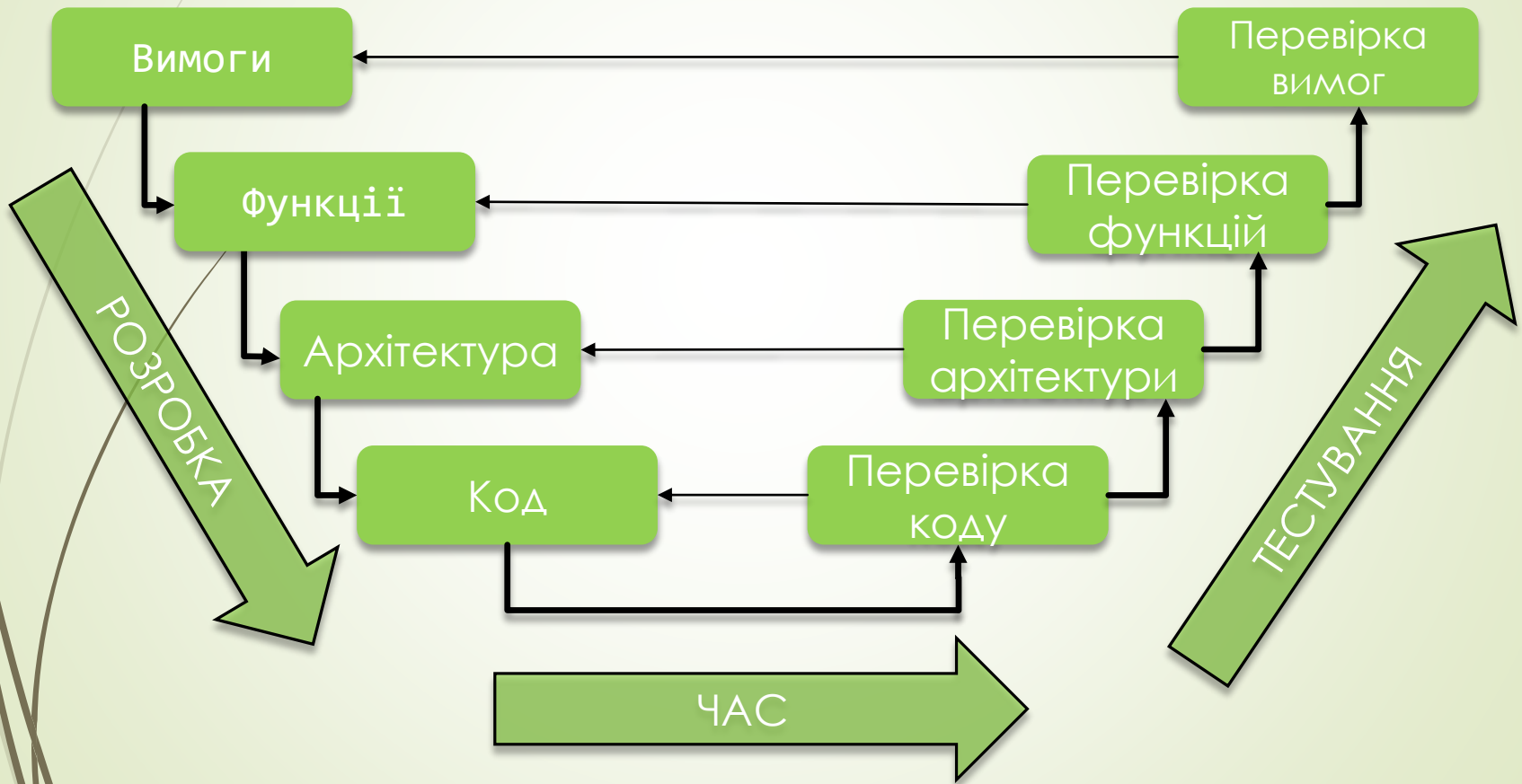
Ітеративне та інкрементне розроблення програм



Класична ітеративна модель

- **При застосуванні даної моделі необхідно враховувати наступні чинники ризику:**
 - вимоги складені незрозуміло для реалізації;
 - всі можливості системи потрібно реалізувати із самого початку;
 - швидко міняються технології і вимоги до системи;
 - обмеження в ресурсному забезпеченні можуть привести до затягування термінів здачі системи в експлуатацію.
- **Дану модель розробки доцільно використати, у випадках коли:**
 - бажано реалізувати деякі можливості системи швидко за рахунок створення проміжного продукту;
 - система розділена на окремі складові частини структури, які можна уявляти як деякий проміжний продукт;
 - можливе збільшення фінансування на розробку окремих частин системи.

V-подібна модель

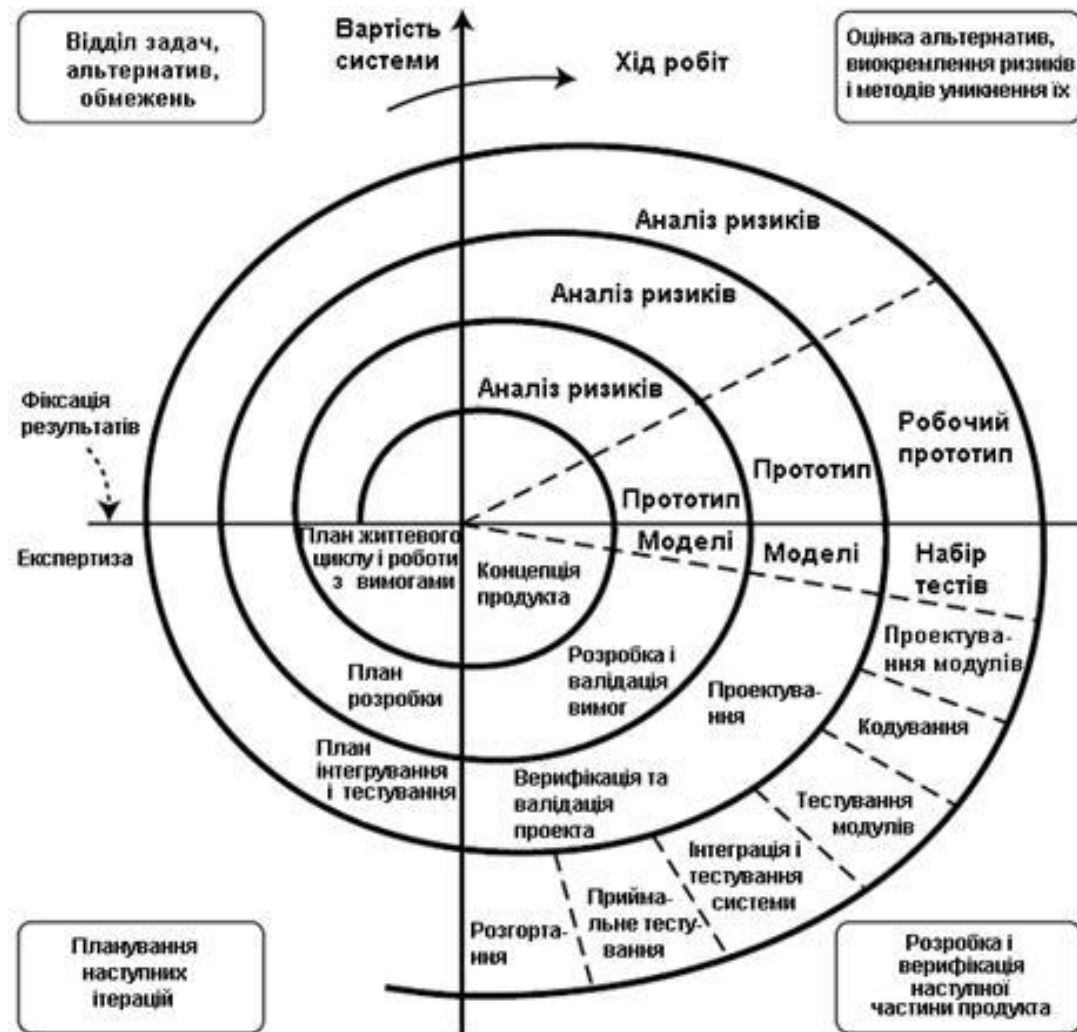


V-подібна модель

Характеристики V-подібної моделі

- Модель була створена для допомоги в плануванні та забезпеченні подальшої можливості тестування системи.
- В моделі особливе значення надається діям, спрямованим на верифікацію й атестацію продукту.
- План приймання кінцевого програмного продукту замовником розробляється на етапі планування, а планування решти системи – на фазах аналізу, розробки проекту тощо.
- Кожна наступна фаза починається лише по завершенню попередньої. Тестування здійснюється після того, як продукт створено. Виявлені помилки спричиняють повернення на попередні етапи життєвого циклу.

Спиральна модель



Спіральна модель

- **Переваги реалізації системи за допомогою цієї моделі:**
- Внесення змін орієнтоване на задоволення потреби користувачів одразу, як тільки буде встановлено, що створені артефакти або елементи документації не відповідають дійсному стану розробки.
- Дана модель ЖЦ допускає аналіз продукту на витку розробки, його перевірку, оцінку правильності та прийняття рішення про перехід на наступний виток або повернення на попередній виток для доопрацювання на ньому проміжного продукту.
- Відмінність цієї моделі від каскадної полягає в можливості багато разів повертатися до процесу формулювання вимог і до повторної розробки версії системи з будь-якого процесу моделі.
- **Недоліки цієї моделі:**
- висловлення вимог замовником носить суб'єктивний характер, вимоги можуть багаторазово уточнюватись протягом розробки ПС і навіть після завершення та випробовування, і часом може з'ясуватися, що замовник «хотів зовсім інше».
- При необхідності внесення змін до системи на кожному витку з метою отримання нової версії системи обов'язково вносяться зміни в заздалегідь зафіксовані вимоги, після чого повертаються на попередній виток спіралі для продовження реалізації нової версії системи з урахуванням усіх змін.

Еволюційна модель



Еволюційна модель

- **При цьому підході враховуються такі чинники ризику:**
 - реалізація всіх функцій системи одночасно може привести до громіздкості;
 - обмежені ресурси зайняті розробкою протягом тривалого часу.
- **Переваги застосування даної моделі ЖЦ:**
 - швидка реалізація деяких функціональних можливостей системи;
 - у системі виділяються окремі частини для реалізації їх у вигляді прототипу;
 - можливість збільшення фінансування системи;
 - можливість зворотного зв'язку із замовником для уточнення вимог;
 - спрощення внесення змін у зв'язку із заміною окремих функцій.

Вибір моделі. На основі вимог

Питання	Каскадна	Спіральна	Еволюційна	Інкрементна
Чи є вимоги добре відомими або такими, що легко визначаються?	Так	Ні	Так	Ні
Чи можуть вимоги визначатись у циклі?	Так	Ні	Так	Так
Чи часто змінюватимуться вимоги в циклі?	Ні	Так	Ні	Ні
Чи потрібно демонструвати вимоги?	Ні	Так	Так	Ні
Чи потрібна перевірка концепції для демонстрації можливостей?	Ні	Так	Так	Ні
Чи будуть вимоги відображати складність програмної системи?	Ні	Так	Ні	Так
Чи володіє вимога функціональними властивостями на ранньому етапі?	Ні	Так	Так	Так

Вибір моделі. Колектив розробників

Питання	Каскадна	Спіральна	Еволюційна	Інкрементна
Чи є проблеми предметної галузі проекту новими для більшості розробників?	Ні	Так	Ні	Ні
Чи є технологія предметної галузі проекту новою для більшості розробників?	Так	Так	Ні	Так
Чи є інструменти проекту новими для більшості розробників?	Так	Так	Ні	Ні
Чи змінюються ролі учасників проекту?	Ні	Так	Ні	Так
Чи можуть розробники проекту навчатись?	Ні	Ні	Так	Так
Чи є структура більш значущою для розробників, ніж гнучкість?	Так	Ні	Ні	Так
Чи буде менеджер проекту суворо відстежувати прогрес команди?	Так	Так	Ні	Так
Чи важлива легкість розподілу ресурсів?	Так	Ні	Так	Так
Чи сприймає колектив рівноправні огляди?	Так	Так	Ні	Так

Вибір моделі. Характеристики користувачів

Питання	Каскадна	Спіральна	Еволюційна	Інкрементна
Чи є присутність користувачів обмежена?	Так	Так	Ні	Так
Чи будуть користувачі знайомі з визначенням програмної системи?	Ні	Так	Ні	Так
Чи будуть користувачі ознайомлені з проблемами предметної галузі?	Ні	Ні	Так	Так
Чи користувачі задіяні на всіх етапах ЖЦ?	Ні	Ні	Так	Ні
Чи відстежуватиме замовник хід проекту?	Ні	Так	Ні	Ні

Вибір моделі. Тип проекту та ризики

Питання	Каскадна	Спіральна	Еволюційна	Інкрементна
Чи буде проект ідентифікувати новий напрямок продукту для організації?	Ні	Так	Ні	Так
Чи буде проект мати системну інтеграцію?	Ні	Так	Так	Так
Чи є проект розширенням існуючого ПЗ?	Ні	Ні	Так	Так
Чи буде фінансування проекту стабільним?	Так	Ні	Так	Ні
Чи очікується тривала експлуатація програмної системи в організації?	Так	Так	Ні	Так
Чи потрібен високий ступінь надійності?	Ні	Так	Ні	Так
Чи буде ПЗ змінюватись непередбаченими методами на етапі супроводу?	Ні	Так	Ні	Так
Чи є графік обмеженим?	Ні	Так	Так	Так
Є повторно використовувані компоненти?	Ні	Так	Так	Ні
Чи є достатніми ресурси?	Ні	Так	Ні	Ні



(づ。♡♡。)*:.。☆