­­МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

­­­

Контрольна робота

з курсу «Технологія створення програмних продуктів»

для студентів базового напрямку 6.08.04 "Комп’ютерні науки"

(заочна форма навчання)

Варіант 10

Виконав студент гр. КНз-3

Чалий Михайло

­­

Львів 2015

# Контрольна робота 1

## Поняття ІТ-проекту. Дати характеристику. Проблемні, провалені й успішні ІТ проекти.

*ІТ-проект* – сукупність процесів створення нового або модифікації існуючого ІТ-продукту, що може включати наступні види робіт: аналіз вимог замовника, формування технічного завдання, проектування та реалізація / придбання, тестування, впровадження / монтажно-налагоджувальні роботи тощо.

Є кілька основних чинників, що визначають кожний конкретний проект: масштаб, складність, терміни реалізації, обмеженість ресурсів, вимоги до якості тощо.

За термінами реалізації розрізняють проекти короткострокові (менше року), середньострокові (1-3 роки) і довгострокові (понад три роки).

За масштабами у міжнародній практиці проекти поділяють на малі (до 10 млн дол.), ***середні*** (10-50 млн дол.), ***великі*** (50- 100 млн дол.) і ***масштабні*** (понад 100 млн дол.).

За складністю розрізняють такі проекти: прості - окремі конкретні проекти з чітко визначеною орієнтацією та масштабом; припускають певні спрощення щодо проектування та реалізації, формування команди проекту тощо; мультипроекти - комплексні проекти, що складаються з простих проектів; мегапроекти - комплексні проекти розвитку регіонів, секторів економіки.

До мегапроектів належать міжнародні проекти, що вирізняються значною організаційною й технічною складністю та високою вартістю, а також великою роллю в економіці й політиці країн, для яких розробляються.

До основних чинників, які необхідно враховувати при розробці мегапроектів, належать такі: розподіл елементів проекту між виконавцями і необхідність координації їхньої діяльності; необхідність аналізу соціально-економічного середовища регіону, країни, де розроблюється проект, і учасників проекту; необхідність виокремлення як самостійної фази розробки концепції проекту; розробка і постійне відновлення плану проекту при його реалізації; необхідність планування на всіх рівнях.

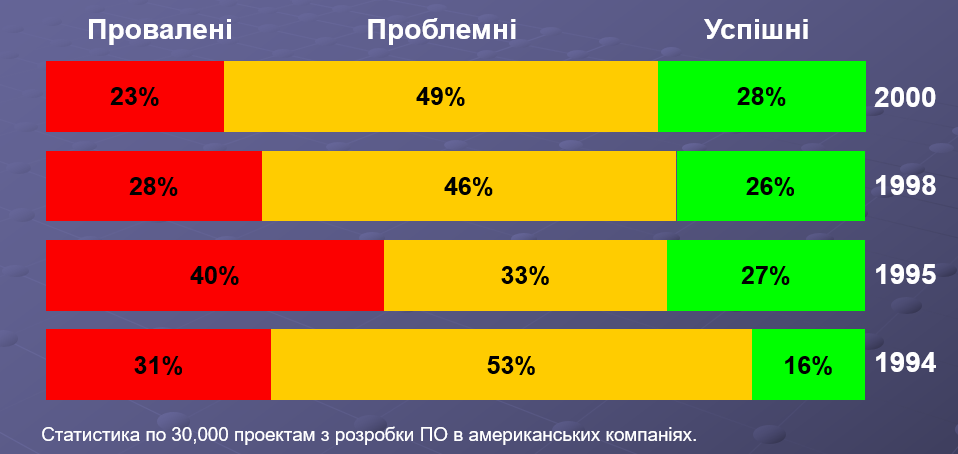
За видами розрізняють проекти комерційні та некомерцій-ні, залежно від мети проекту: отримати прибуток або інший ефект.

За характером і сферою діяльності виокремлюють такі проекти: промислові, економічні, організаційні, дослідницькі, соціальні тощо.

З метою розв'язання всіх необхідних задач за проектом використовують сучасні програмні продукти.

За результататми проекту:

* ***У*спішні** проекти – вчасно і в рамках бюджету було виконано весь намічений фронт робіт.
* **Проблемні** – не вклались у строки, перевищили бюджет і/або зробили не все, що потрібно.
* **Провалені** – не були доведені до кінця.



## Поняття “якості” програмного продукту. Критерії якості. Навести приклади і пояснити.

**Якість програмного забезпечення** — характеристика програмного забезпечення, ступінь відповідності ПЗ до вимог. При цьому вимоги можуть трактуватись по-різному, що породжує декілька незалежних визначень терміну. Якість ПЗ – набір властивостей продукту (сервісу або програм), що характеризують його здатність задовольнити встановлені або передбачувані потреби замовника. Поняття якості має різні інтерпретації залежно від конкретної програмної системи і вимог до неї.

Деякі критерії:

* Прочитність коду
* Легкість підтримки, тестування, відлагодження, виправляння помилок, рефакторингу та портування
* Низька складність коду
* Коректність обробки винятків

## Прототипіювання при створенні ПЗ. Зобразити алгоритм і пояснити. Переваги і недоліки.

У програмній інженерії прототипування означає створення часткових програмно-апаратних реалізацій моделей програмного забезпечення. Варто зауважити, що у програмну інженерію термін «прототип» потрапив з інших сфер інженерії (наприклад, машинобудування). Проте якщо в інших сферах прототипування – усталений процес, етап життєвого циклу виробництва, який працює за перевіреною десятиліттями схемою, то в програмній інженерії, що бурхливо розвивається, місце прототипування залишається неоднозначним.



Використання прототипів під час моделювання та розроблення програмних систем базується на низці переваг, які надає прототипування. Водночас прототипування як модель життєвого циклу ПЗ нерідко зазнає критики, обґрунтованої певними недоліками. Переваги й недоліки прототипування наведено нижче.

***Переваги прототипування***

*Мінімізація часових і фінансових витрат*: прототипування може підвищити якість вимог і специфікацій, що надаються розробникам. Раннє визначення дійсних вимог і потреб користувача спонукає до швидшої та дешевшої розробки програмного забезпечення.

*Вдосконалене залучення кінцевого користувача до процесу створення*: наявність прототипу, досліджуваного користувачем, запобігає багатьом непорозумінням, які зазвичайвиникають між програмістами та замовниками.

*Навчання користувача:* прототип може слугувати для навчання користувачів роботі зі складною системою, доки сама система ще перебуває у розробленні.

***Недоліки прототипування***

*Недостатній аналіз:* концентрація уваги на обмеженому прототипі може завадити коректному аналізу повного проекту. Це може призвести до вибору не найліпших рішень, підготовки неповних специфікацій або перетворення обмежених прототипів у неправильно спроектовані системи, які важко підтримувати.

*Непорозуміння з користувачем, який не розуміє різниці між прототипом та завершеною системою*: якщо прототип містить тільки візуальну оболонку, замовникові може бутиважко пояснити, що створення візуальноїоболонки становить значно менше часу, ніжвтілення всієї внутрішньої логіки, передбаче ної специфікаціями.

*Неправильне використання часу розробником:* ключова особливість прототипу – швидкість його створення. Якщо побудова прототипу та поліпшення його функціональностітривалі, відбувається спотворення основноїпарадигми прототипування, результатом чогоє проблеми з часовими рамками проекту.

*Зацикленість на прототипі:* після виготовлення прототипу в програмістів виникає спокуса подальшого його використання як бази проекту і розробка зводиться до надбудови модулів над моделлю. Такий підхід стає причиною некоректних архітектурних рішень.

## Розрахунок та планування часу виконання ІТ проекту. Правило розподілу витрат проекту.

**Оцінка тривалості робіт** включає визначення кількості робочих періодів, яка ймовірніше за все знадобиться для завершення будь-якої певної роботи. Особа чи група осіб з команди проекту, яка найкраще обізнана з конкретною роботою, що має бути виконана, повинна дати таку оцінку або принаймні підтвердити вже існуючу.

Оцінка кількості робочих періодів, необхідних для завершення роботи, часто вимагає також обліку й усього часу, що минув. Загальна тривалість проекту може бути оцінена з допомогою методів і засобів, що розглядаються тут, але краще її визначати як результат розробки календарного плану.

**Методи та засоби оцінки тривалості робіт**

*1. Висновок експерта.*. Часто буває важко оцінити тривалість робіт, оскільки на них впливає велика кількість чинників (наприклад, рівень використання ресурсів, продуктивність ресурсів). Висновок експерта, що грунтується на інформації з архіву, повинен використовуватися скрізь, де є на це можливість. У противному разі оцінки набувають невизначеності й стають ризикованими.

*2. Оцінка на основі аналогів.* Оцінка на основі аналогів, або оцінка зверху-вниз, означає використання фактичної тривалості попередньої аналогічної роботи як оцінки тривалості майбутньої роботи. Вона часто використовується для оцінки тривалості проекту, коли про нього є небагато детальної інформації (наприклад, на його ранніх фазах). Оцінка на основі аналогів є однією з форм висновку експерта.

Оцінка на основі аналогів найбільш надійна, коли (а) попередні проекти схожі не тільки за формою, а й за змістом, і коли (Ь) особи, що виконують цю роботу, мають необхідний досвід.

*3. Моделювання.* Моделювання включає розрахунок великої кількості тривалостей з певним набором допущень. Найбільш відомим є метод «Монте-Карло», в якому розподіл імовірних результатів визначається по кожній роботі і використовується для розрахунку розподілу ймовірних результатів по проекту в цілому

Результати оцінки тривалості робіт

*1. Оцінки тривалості робіт.* Оцінки тривалості робіт - це кількісні оцінки ймовірної кількості робочих періодів, необхідних для завершення роботи. Вони завжди повинні включати вказівку на діапазон можливих результатів. Наприклад:

* 2 тижні +- 2 дні для вказівки, що робота займатиме як мінімум 8 днів, але не більш ніж 12.
* 15-відсоткова ймовірність перевищення тритижневого строку для вказівки високої імовірності - 85 відсотків - що робота займатиме 3 тижні або менше.

## Декомпозиція ПЗ на модулі. Пояснити переваги і недоліки модульних ПЗ

Виділення в складі системи підсистем і елементів називають декомпозицією системи.

Декомпозиція системи залежить:

* від типу системи (сільськогосподарське підприємство, переробне підприємство, сервісне підприємство і т. ін.);
* від мети діагностики (для формулювання місії і цілей підприємства, для оцінки конкурентоспроможності тощо);
* від методики аналізу тощо.

Загальні критерії декомпозиції підсистем:

* Суттєвий вплив на цілі і кінцеві результати системи; о Відображення інтегрованого впливу своїх елементів; - - Зв'язок із загальносистемними характеристиками, які мають вплив на досягнення цілей за допомогою усієї системи;
* Необхідність створення підсистем за тими ознаками, що чітко проявляють функціональний зв'язок між підсистемами і системою в цілому;
* Функції підсистеми реалізуються лише через взаємодію з іншими підсистемами і елементами.

## Контроль структури програми

Для контролю структури програмного забезпечення використовується суміжний контроль і ручний контроль.

Суміжний контроль архітектури програмного забезпечення зверху - це її контроль розроблювачами зовнішнього опису, розроблювачами специфікації якості і розроблювачами функціональної специфікації. Суміжний контроль архітектури програмного забезпечення знизу - це її контроль потенційними розроблювачами програмних підсистем, що входять до складу програмного забезпечення відповідно до розробленої архітектури.

Ручний контроль архітектури програмного забезпечення виконується аналогічно ручного контролю функціональної специфікації, тільки метою цього контролю є перевірка взаємодії між програмними підсистемами. Так само як і у випадку ручного контролю функціональної специфікації програмного забезпечення повинні бути спочатку підготовлені тести. Потім група розроблювачів повинна для кожного такого тесту контролювати роботу кожної програмної підсистеми, що входить до складу програмного забезпечення. При цьому роботу кожної підсистеми контролює один який-небудь розроблювач (не автор архітектури), ретельно виконуючи усі взаємодії цієї підсистеми з іншими підсистемами (точніше, з розроблювачами, що їх імітують) відповідно до розробленої архітектури програмного забезпечення. Тим самим забезпечується контрольоване функціонування програмного забезпечення у цілому в рамках архітектури, що перевіряється.

# Контрольна робота №2

## Поясніть порядок і принципи об’днання ролей в моделі команди MSF. Наведіть приклад.

***MSF заснований на постулаті про шість якісних цілях, досягнення яких визначає успішність проекту.*** Ці цілі обумовлюють модель проектної групи. У той час як за успіх проекту відповідальна вся команда, кожен з її рольових кластерів, що визначаються моделлю, асоційований з однієї зі згаданих шести цілей і працює над її досягненням.

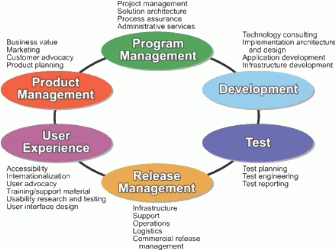
Модель проектної групи MSF підкреслює важливість побудови рольових кластерів відповідно до потреб бізнесу. Угруповання пов'язаних областей компетенції, кожна з яких має свою специфіку, забезпечує хорошу збалансованість команди.

Чітке визначення цілей підвищує рівень відповідальності і сприяє кращому їх сприйняття проектною командою, що негайно позначається найкращим чином на якості продукту, що випускається.

Оскільки кожна з цілей однаково необхідна для успішності проекту, всі ролі знаходяться в рівноправних партнерських взаєминах з рівною значимістю при прийнятті рішень.

Використання рольових кластерів не має на увазі і не нав'язує ніякої спеціальної структури організації або обов'язкових посад (ріс.4.15). Адміністративний склад ролей може широко варіюватися в різних організаціях і проектних групах. Найчастіше ролі розподіляються серед різних підрозділів однієї організації, але іноді частина їх відводиться спільноті споживачів або зовнішнім по відношенню до організації консультантам і партнерам.

Ключовим моментом є чітке визначення працівників, відповідальних за кожен рольової кластер, їх функцій, відповідальності та очікуваного вкладу в кінцевий результат

[](http://technet.microsoft.com/en-us/library/Bb497039.ump0202_big(l=en-us).gif)

**Функції та області компетенції рольових кластерів**

|  |  |  |
| --- | --- | --- |
| **Рольової кластер**  **Мета** | **Область компетенції** | **Функції** |
| Управління продуктом  Мета: Задоволені замовники | Маркетинг  Бізнес- віддача (бізнес-пріоритети)  Представлення інтересів замовника  Планування продукту | Виступає в ролі представника замовника  Формує спільне бачення / рамки проекту  Організовує роботу з вимогами замовника  Розвиває сфери застосування в бізнесі  Формує очікування замовника  Визначає компроміси між параметрами «можливості продукту / час / ресурси»  Організовує маркетинг, PR та євангелізацію  Розробляє, підтримує і виконує план комунікацій |
| Управління програмою  Мета: Досягнення результату в рамках проектних обмежень | Управління проектом  Вироблення архітектури рішення  Контроль виробничого процесу  Адміністративні служби | Управляє процесом розробки з метою отримання готового продукту у відведені терміни  Формулює специфікацію продукту і розробляє його архітектуру  Регулює взаємовідносини і комунікацію всередині проектної групи  Стежить за тимчасовим графіком проекту і готує звітність про його стан  Проводить у життя важливі компромісні рішення  Розробляє, підтримує і виконує зведений план і календарний графік проекту  Організує управління ризиками |
| Розробка  Мета: Створення продукту відповідно до специфікації | Технологічне консультування  Проектування і здійснення реалізації  Розробка додатків  Розробка інфраструктури | Визначає деталі фізичного дизайну  Оцінює необхідні час і ресурси  на реалізацію кожного елемента дизайну  Розробляє або контролює розробку елементів  Готує продукт до впровадження  Консультує команду з технологічних питань |
| Тестування  Мета: Схвалення випуску продукту тільки лише після того, як всі дефекти виявлені і улагоджені | Планування тестів  Розробка тестів  Звітність по тестах | Забезпечує виявлення всіх дефектів  Розробляє стратегію та плани тестування  Здійснює тестування |
| Задоволення споживача  Мета: Підвищення ефективності користувача, збільшення споживчої цінності продукту | Забезпечення технічної підтримки  Навчання  Ергономіка  Графічний дизайн  Інтернаціоналізація  Загальнодоступність (забезпечення можливості роботи для користувачів з обмеженими фізичними можливостями) | Представляє інтереси споживача в ко  манде  Організовує роботу з вимогами користувача  Проектує і розробляє системи підтримки продуктивності  Визначає компроміси, пов'язані  до зручності використання і споживчим якостям продукту  Визначає вимоги до системи допомоги та її зміст  Розробляє навчальні матеріали та здійснює навчання користувачів |
| Управління випуском  Мета: Безпроблемне впровадження та супровід продукту | Інфраструктура  Супровід  Бізнес- процеси  Управління випуском готового продукту | Представляє інтереси відділів постачання  та обслуговування продукту  Організовує постачання проектної групи  Організовує впровадження продукту  Виробляє компроміси в керованості і зручності супроводу продукту  Організовує супровід та інфраструктуру поставки  Організовує логістичне забезпечення |

## Характеристика фази “вироблення концепції” в моделі процесів MSF: віхи, завдання і результати

На етапі «вироблення концепції» вирішуються такі основні завдання: оцінка існуючої ситуації, визначення складу команди, структури проекту, бізнес-цілей, вимог і профілів користувачів; розробка концепції рішення і оцінка ризику. Встановлюються дві проміжні віхи: "Організовано кістяк команди" і "Створено загальну картину рішення".

|  |  |
| --- | --- |
| Основні завдання | Owners Власники |
| **Setting up a team Налаштування команду**  Senior management determines that a project is viable and selects a Program Manager whose first task is to assemble a project team that represents all six MSF team roles. Вище керівництво визначає, що проект є життєздатним і вибирає менеджер програму, в якій перша задача полягає в зібрати команду проекту, який представляє всі шість MSF командних ролей. | Product Management Управління продуктом  Program Management Управління програмою |
| Defining the project structure **Визначення структури проекту**  The team creates the project structure, which describes the administrative structure for the project team going into the Planning Phase. Команда створює структуру проекту, в якому описується адміністративна структура проектної команди відбувається у фазі планування. It includes standards the team will use to manage and support the project. Вона включає в себе стандарти команда буде використовувати для управління та підтримки проекту. | Program Management Управління програмою |
| **Defining the business goals Визначення бізнес-цілей**  The team identifies the business problems or opportunities in order to determine the objectives for the solution. Команда визначає бізнес-завдань і можливостей для того, щоб визначити цілі для рішення. | Product Management Управління продуктом |
| Assessing the current situation **Оцінка поточної ситуації**  The team assesses the current state of the business and performs a gap analysis to help identify a path toward the desired state of the business. Команда оцінює поточний стан бізнесу і виконує аналіз прогалин, щоб допомогти визначити шлях до бажаного стану бізнесу. | Program Management Управління програмою |
| Creating the vision and defining the scope for the project **Створення бачення та визначення сфери для проекту**  The team creates a shared and clearly articulated vision statement that guides the team toward its business goals. Команда створює загальний і чітко сформульованої заяву про бачення, яка направляє команду до її бізнес-цілей. The team also identifies the scope of the project by defining what will and will not be included. Команда також визначає масштаби проекту, визначивши, що буде і не буде включений. | Product Management Управління продуктом  Program Management Управління програмою |
| **Defining high-level requirements and user profiles Визначення вимог високого рівня і профілів користувачів**  The team determines the needs of each key stakeholder, sponsor, and end user in order to provide input regarding the formation of the solution concept, to provide criteria for evaluating the vision/scope document, and to provide a basis for more detailed requirements in the Planning Phase. Команда визначає потреби кожної ключової зацікавленої сторони, спонсора, і кінцевого користувача для того, щоб внести вклад у відношенні формування концепції рішення, щоб забезпечити критерії для оцінки документ бачення / комплект, і створюють основу для більш детальних вимог в етап планування. | Product Management Управління продуктом  User Education Освіта користувача |
| Developing the solution concept **Розробка концепції рішення**  The team transforms the high-level requirements into an initial concept of how the solution solves the business problem. Команда перетворює вимоги високого рівня в початкової концепції, як рішення вирішує бізнес-завдання. The solution concept serves as a baseline and sets the stage for the more formal design of the solution. Концепції рішення служить в якості основи і створює основу для більш формального проекту рішення. | Program Management Управління програмою |
| **Assessing risk Оцінка ризику**  The team proactively identifies project risks and creates plans to deal with them during the Envisioning Phase. Команда активно ідентифікує проектні ризики і створює плани по боротьбі з ними під час фази виробітку. It continues this recurring process throughout the project. Він як і раніше Це повторюється процес протягом всього проекту. | Program Management Управління програмою |
| Closing the Envisioning Phase **Закриття фази вироблення**  The team completes the approval process for the Vision/Scope Approved Milestone by formalizing the documentation that records the results of its tasks and presenting it to management for approval. Команда завершує процес затвердження для бачення / обсягах, затверджених Milestone шляхом формалізації документацію, яка записує результати своїх завдань та подання його керівництву для затвердження. | Project team Команда проекту |

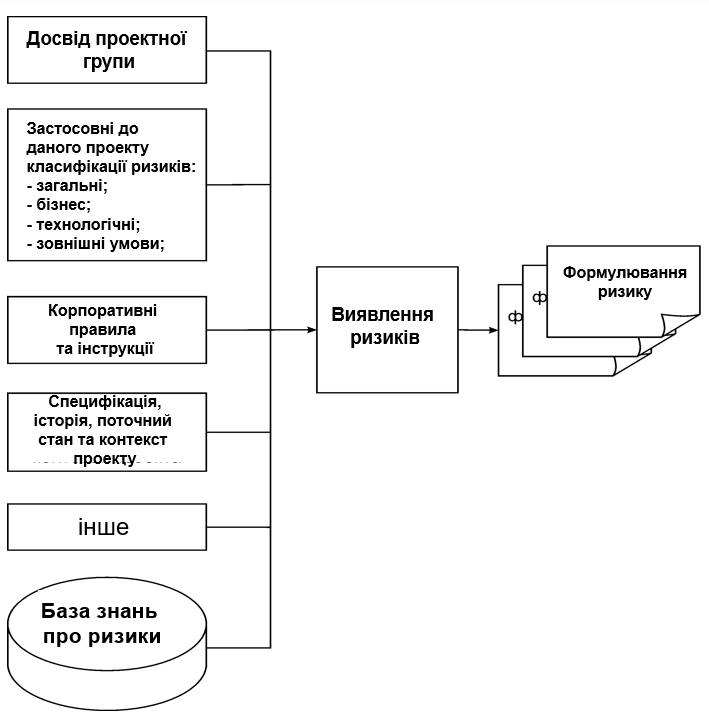
#### Ключові результати від фази вироблення

Нижче перелік для результатів та їх зміст, що команда створює під час фази виробітку:

* Бачення / документ сфера
  + Проблема заяви та бізнес-цілей
  + Огляд існуючих процесів
  + Вимоги високого рівня
  + Профілі користувачів виявлення тих, хто виграє від рішення
  + Заява бачення і визначення сфера
  + Концепції вирішення з викладенням підходу, команда прийме планувати проект
  + Дизайнерське рішення стратегії
* Структура документа проекту
  + Опис і відображення всіх ролей MSF команди
  + Структуру і стандарти технологічного процесу проекту для команди, щоб слідувати
* Документ оцінки ризиків
  + Попередня оцінка ризику
  + Список найкращих виявлених ризиків
* Оглядовий доповідь Milestone
  + Звіт про хід виконання проекту Член команди
  + Команда доповідь керівник проекту

## Характеристика кроку “виявлення ризиків” в процесі управління ризиками MSF

Важливо своєчасно виявити ризики ІТ -проекту, бажано на початкових фазах життєвого циклу, встановити джерела (люди, процеси, технології та зовнішні умови виконання проекту), умови виникнення ризиків. Велике значення мають досвід членів проектної групи, класифікація ризиків, корпоративні правила та інструкції протидії ризикам, база знань про ризики ІТ -проектів.



## Наведіть та поясніть ключові концепції, на яких грунтується дисципліна управління проектами MSF

1. Команда соратників.

Концепція "команди соратників" (teem of peers) означає рівноправне положення кожної з ролей в команді. Це сприяє вільному спілкуванню, збільшує командну відповідальність і зумовлює рівну важливість кожної з шести якісних цілей. Щоб досягти успіху в рамках команди соратників (команди рівних учасників), кожен з її членів, незалежно від ролі, повинен нести відповідальність за якість продукту, розуміти інтереси замовника і суть вирішуваної бізнес-задачі.

1. Сфокусованість на потребах замовника.

Задоволення потреб замовника – головний пріоритет будь-якої добре працюючої проектної групи. Концентрація на потребах замовника (customer-focused mindset) означає обов'язкове розуміння його бізнес-задач і прагнення до їх вирішення з боку команди. Одним із способів визначення успіху такої уваги до замовника є **здатність відстежити зв'язок кожного елементу в дизайні системи з відповідним йому початковою вимогою замовника** або користувача (trace each feature in the design back to а customer or user requirement). Іншим ключовим моментом в задоволенні потреб замовника є його активна участь в проектуванні рішення і отримання його відгуків в ході процесу розробки. Це дозволяє проектній групі і замовникові успішно погоджувати свої очікування і потреби.

1. Націленість на кінцевий результат.

Неважливо, чи займаєтеся ви, подібно до співробітників майкрософту, виробництвом "коробкового" ПО або розробляєте програми для внутрішніх цілей вашого підприємства. Важливо, як ви відноситеся до результатів своєї індивідуальної праці, чи сприймаєте ви їх як продукт.

1. Установка на відсутність дефектів.

В успішній команді кожен співробітник відчуває відповідальність за якість продукту. Вона не може бути делегована одним членом команди іншому або ж одним ролевим кластером іншому. Відповідно, кожен член команди повинен представляти інтереси замовника, враховуючи в ході розробки продукту його споживчі якості.

1. Прагнення до самовдосконалення.

Прагнення до самовдосконалення (willingness to learn) – це прихильність ідеї невпинного саморозвитку за допомогою накопичення досвіду і обміну знаннями. Воно дозволяє членам проектної групи отримувати користь з негативного досвіду зроблених помилок, так само як і відтворювати успіхи, використовуючи перевірені методи роботи інших людей. Проведення по закінченню основних фаз проекту відкритих обговорень його стану і доброзичливий, але об'єктивний аналіз проекту після його закінчення – це ключові компоненти моделі процесу MSF. Проектні групи, що виділяють час на аналіз результатів своєї роботи і витягання з них уроків, створюють базу для постійного самовдосконалення і довготривалого успіху. Крім того, Майкрософт успішно розвиває культуру самовдосконалення, включаючи аналіз уроків, що витягують, і обмін знаннями в плани індивідуальної роботи співробітників.

1. Зацікавлені команди працюють ефективно.

Команди з низькою мотивацією програють по двох причинах. На індивідуальному рівні їх члени не працюють з повною віддачею, що веде до зниження якості роботи і продуктивності праці. Крім цього, увага працівників зосереджена на вузьких цілях, і вони не піклуються про вплив своєї діяльності на роботу колег. Обидва ці чинника роблять істотний вплив на IT-проекти, розробка яких завжди має на увазі високий рівень завдань, що стоять перед колективом, і інтелектуальну складність підходів до їх рішення.

## Розкрийте і поясніть зміст кроку “Оцінювання”в процесі управління підготовкою MSF

MSF включає процес досягнення рівня знань, умінь і здібностей, необхідного для створення і управління проектами та ІТ рішеннями. Цей процес складається з чотирьох кроків: 1) визначення (define), 2) оцінювання (assess), 3) коригування (change) і 4) осмислення (evaluate). Кожен крок включає серію завдань, що допомагають досягнути наступної віхи.

Крок ***"Оцінювання"*** процесу ***Управління підготовкою*** служить для визначення того, яким кваліфікаційним вимогам задовольняють (або не задовольняють) у нинішній момент співробітники організації. На цьому кроці починається аналіз кваліфікаційних вимог різних робочих ролей, що дозволяє визначити необхідні професійні навики співробітників. Потім бажана кваліфікація співробітників зіставляється з тією, що існує на даний момент. Це вказує напрями вдосконалення професійного рівня, що дозволяє почати розробку планів навчання. Залежно від числа робочих ролей, потрібних для успішного використання ІТ-технології, можуть бути вибрані різні оцінки:

* групи кваліфікаційних вимог, що встановлюють професійні навики;
* методики атестації;
* учбові плани.

Завдання цього кроку включають:

* оцінку знань, умінь, здібностей;
* аналіз невідповідностей;
* створення планів навчання.

Результатами цього кроку є:

* отримані оцінки і виявлені невідповідності;
* плани навчання.