 [Translated from Russian to Ukrainian - www.onlinedoctranslator.com](https://www.onlinedoctranslator.com/en/?utm_source=onlinedoctranslator&utm_medium=docx&utm_campaign=attribution)

## 4. Управління проектами

**Цілі**

Мета цієї глави – огляд процесів управління програмними проектами. Прочитавши цей розділ, ви повинні:

* розуміти різницю між управлінням програмними проектами та управлінням інженерними проектами інших типів;
* знати основні завдання, які стоять перед керівником програмного проекту;
* розуміти значимість та роль етапу планування проекту серед усіх етапів процесу створення ПЗ;
* знати графічні методи представлення графіків робіт;
* мати уявлення про типи ризиків, що виникають при реалізації програмних проектів, та процеси управління цими ризиками.

Проблеми управління програмними проектами вперше виявилися у 60-х – початку 70-х, коли провалилися багато великих проектів із розробки програмних продуктів. Було зафіксовано затримки у створенні ПЗ, воно було ненадійним, витрати на розробку в кілька разів перевершували початкові оцінки, створені програмні системи часто мали низькі показники продуктивності [60]. Провали цих проектів зумовлювалися не лише некомпетентністю керівників та програмістів. Навпаки, у цих великих пошукових проектах брали участь люди, рівень кваліфікації яких був явно вищим за середній. Причини провалів коренилися в підходах, які використовувалися в управлінні проектами. Методика, що застосовується, була заснована на досвіді управління технічними проектами і виявилася неефективною при розробці програмного забезпечення.

Тут важливо зрозуміти різницю між професійною розробкою ПЗ та аматорським програмуванням. Необхідність управління програмними проектами випливає з того "сумного" факту, що створення професійного ПЗ завжди є суб'єктом бюджетної політики організації, де воно розробляється, і має тимчасові обмеження. Робота керівника програмного проекту за великим рахунком полягає в тому, щоб гарантувати виконання цих бюджетних та тимчасових обмежень з урахуванням бізнес-цілей організації щодо ПЗ, що розробляється.

Менеджери проектів мають спланувати всі етапи розробки програмного продукту. Вони також повинні контролювати хід виконання робіт та дотримання всіх необхідних стандартів. Постійний контроль за ходом виконання робіт необхідний для того, щоб процес розробки не виходив за тимчасові та бюджетні обмеження. Хороше керування не гарантує успішного завершення проекту, але погане керування обов'язково призведе до його провалу. Це може виразитись у затримці термінів здачі готового ПЗ, у перевищенні кошторисної вартості проекту та у невідповідності готового ПЗ специфікації вимог.

Керівники програмних проектів виконують таку ж роботу, що й керівники технічних проектів. Разом з тим процес розробки ПЗ істотно відрізняється від процесів реалізації технічних проектів, що породжує певні складнощі в управлінні програмними проектами. Наведемо невеликий перелік цих відмінностей.

1. *Програмний продукт є нематеріальним.*Менеджер суднобудівного проекту чи проекту спорудження будівлі бачить результати виконання свого проекту. Якщо реалізація проекту відстає від графіка, це також видно на власні очі, оскільки частина конструкції не завершена. На противагу цьому програмне забезпечення нематеріальне. Його не можна побачити чи торкнутися. Менеджер програмного проекту не бачить процес "зростання" ПЗ, що розробляється. Він може покладатися лише документацію, яка фіксує процес розробки програмного продукту.

2. *Немає стандартних процесів розробки ПЗ.*На сьогоднішній день немає чіткої залежності між процесом створення ПЗ і типом створюваного програмного продукту. Інші технічні дисципліни мають тривалу історію, процеси розробки технічних виробів багаторазово випробувані та перевірені. Процеси створення більшості технічних систем добре вивчені. Вивченням процесів створення ПЗ фахівці займаються лише кілька останніх років. Тому поки що не можна точно передбачити, на якому етапі процесу розробки програмного забезпечення можуть виникнути проблеми, що загрожують усьому програмному проекту.

3. *Великі програмні проекти – це часто "одноразові" проекти.*Великі програмні проекти зазвичай значно відрізняються від проектів, реалізованих раніше. Тому, щоб зменшити невизначеність у плануванні проекту, керівники проектів повинні мати дуже великий практичний досвід. Але постійні технологічні зміни у комп'ютерній техніці та комунікаційному обладнанні знецінюють попередній досвід. Знання та навички, накопичені досвідом, можуть не потрібні в новому проекті.

Наведене вище може призвести до того, що реалізація проекту вийде з тимчасового графіка або перевищить бюджетні асигнування. Програмні системи найчастіше виявляються новинками як в "ідеологічному", так і в технічному плані. Технічні проекти, які є інноваційними (наприклад, нова транспортна система) також часто порушують тимчасові графіки робіт. Тому, передбачаючи можливі проблеми у реалізації програмного проекту, слід завжди пам'ятати, що багатьом із них властиво виходити за межі тимчасових та бюджетних обмежень.

Управління програмними проектами – тема дуже велика, яку неможливо висвітлити в одному розділі. Тому в цьому розділі дається лише введення в цю тему і розглядається три основні процеси, що виконуються в рамках управління проектами, а саме: планування проекту, складання графіка робіт та управління ризиками. У частині VI представлені інші аспекти управління проектами розробки програмного забезпечення, у тому числі управління персоналом, оцінювання вартості проекту та управління якістю програмного забезпечення.

### 4.1. Процеси управління

Неможливо описати та стандартизувати всі роботи, які виконує менеджер проекту з створення ПЗ. Ці роботи дуже істотно залежать від організації, де виконується розробка, і від типу створюваного програмного продукту. Але в будь-якому випадку більшість менеджерів відповідальні за виконання всіх або деяких із наведених нижче процесів управління.

• Написання пропозицій щодо створення ПЗ.

• Планування та складання графіка робіт зі створення ПЗ.

• Оцінювання вартості проекту.

• Контроль за ходом виконання робіт.

• Підбір персоналу.

• Написання звітів та подань.

Перша стадія програмного проекту може складатися з написання пропозицій щодо реалізації цього проекту. Пропозиції повинні містити опис цілей проектів та способів їх досягнення. Вони також зазвичай включають оцінки фінансових і тимчасових витрат на виконання проекту. При необхідності тут можуть наводитися обгрунтування передачі проекту виконання сторонньої організації чи команді розробників.

Написання пропозицій – дуже відповідальна робота, оскільки багатьом організацій питання у тому, проект виконуватиметься самої організацією чи розроблятися за контрактом сторонньої компанією, є критичним. Неможливо дати будь-які рекомендації щодо написання речень, багато чого тут залежить від досвіду менеджера. Ейрон (Агоп) [12] вважає цю роботу менеджера однією з найважливіших серед інших виконуваних ним робіт.

На етапі планування проекту визначаються процеси, етапи та отримані кожному з них результати, які мають призвести до виконання проекту. Реалізація цього плану призведе до досягнення цілей проекту. Визначення вартості проекту безпосередньо пов'язане з його плануванням, оскільки оцінюються ресурси, потрібні до виконання плану. Ці питання обговорюються в цьому розділі, а також у розділі 23.

Контроль за ходом виконання робіт (моніторинг проекту) – це безперервний процес, який триває протягом усього терміну реалізації проекту. Менеджер повинен постійно відстежувати хід реалізації проекту та порівнювати фактичні та планові показники виконання робіт із їх вартістю. Хоча багато організацій мають механізми формального моніторингу робіт, досвідчений менеджер може скласти ясну картину стадії розвитку проекту просто шляхом неформального спілкування з розробниками.

Неформальний моніторинг часто допомагає виявити потенційні проблеми, які явно можуть виявитися пізніше. Наприклад, щоденне обговорення ходу виконання робіт може виявити окремі недоробки у створюваному програмному продукті. Замість очікування звітів, в яких буде відображено факт "пробуксування" графіка робіт, менеджер може обговорити з фахівцями програмістські проблеми, що плануються, і не допустити зриву графіка робіт.

Протягом реалізації проекту зазвичай відбувається кілька формальних контрольних перевірок ходу виконання робіт із створення ПЗ. Такі перевірки мають дати загальну картину ходу реалізації проекту загалом і показати, наскільки вже розроблена частина програмного забезпечення відповідає цілям проекту.

Час виконання великих програмних проектів може тривати кілька років. Протягом цього часу цілі та наміри організації, що замовила програмний проект, можуть суттєво змінитись. Може виявитися, що програмний продукт, що розробляється, став уже непотрібним або вихідні вимоги до створюваного ПЗ просто застаріли і їх необхідно кардинально змінювати. У такій ситуації керівництво організації-розробника може ухвалити рішення про припинення розробки ПЗ або про зміну проекту в цілому для того, щоб врахувати цілі та наміри організації-замовника, що змінилися.

Керівники – менеджери проектів зазвичай зобов'язані самі вибирати виконавців для своїх проектів. В ідеальному випадку професійний рівень виконавців повинен відповідати роботі, яку вони виконуватимуть у ході реалізації проекту. Однак у багатьох випадках менеджери повинні покладатися на команду розробників, яка є далекою від ідеальної. Така ситуація може бути спричинена наступними причинами.

1. Бюджет проекту не дає змоги залучити висококваліфікований персонал. У такому разі за меншу платню залучаються менш кваліфіковані спеціалісти.

2. Бувають ситуації, коли неможливо знайти фахівців необхідної кваліфікації як у самій організації-розробнику, так і поза нею. Наприклад, в організації "найкращі люди" можуть бути зайняті в інших проектах.

3. Організація хоче підвищити професійний рівень своїх працівників. У цьому випадку вона може залучити до участі в проекті недосвідчених або недостатньо кваліфікованих працівників, щоб вони набули необхідного досвіду та повчилися у досвідченіших фахівців.

Таким чином, майже завжди підбір фахівців для виконання проекту має певні обмеження та не є вільним. Водночас необхідно, щоб хоча б кілька членів групи розробників мали кваліфікацію та досвід, достатні для роботи над цим проектом. В іншому випадку неможливо уникнути помилок у розробці ПЗ. Проблеми створення команд розробників та добору персоналу докладніше розглядаються у розділі 22.

Менеджер проекту зазвичай зобов'язаний надсилати звіти про його виконання як замовнику, і підрядним організаціям. Це мають бути короткі документи, що ґрунтуються на інформації, що витягується з докладних звітів про проект. У цих звітах має бути та інформація, яка дозволяє чітко оцінити рівень готовності створюваного програмного продукту.

### 4.2. Планування проекту

Ефективне управління програмним проектом залежить від правильного планування робіт, необхідні його виконання. План допомагає менеджеру передбачити проблеми, які можуть виникнути на будь-яких етапах створення ПЗ, та розробити превентивні заходи для їх попередження чи вирішення. План, розроблений на початковому етапі проекту, розглядається всіма його учасниками як керівний документ, виконання якого має призвести до успішного завершення проекту. Цей початковий план має максимально докладно описувати всі етапи реалізації проекту.

Структура плану створення ПЗ розглядається у розділі 4.2.1. Тут лише зазначимо, що, окрім розробки плану проекту, менеджер лягає обов'язок розробки інших видів планів. Ці види планів коротко описані у табл. 4.1 та детально обговорюються у відповідних розділах книги.

**Таблиця**4.1. Види планів

|  |  |
| --- | --- |
| **План** | **Опис** |
| План якості | Описує стандарти та заходи щодо підтримки якості ПЗ, що розробляється (глава 24) |
| План атестації | Описує засоби, ресурси та перелік робіт, необхідних для атестації програмної системи (глава 19) |
| План управління конфігурацією | Описує структуру та процеси управління конфігурацією (глава 29) |
| План супроводу ПЗ | Пропонує план заходів, потрібних для супроводження програмного забезпечення у процесі його експлуатації, а також розрахунок вартості супроводу та необхідні для цього ресурси (глава 27) |
| План з управління персоналом | Описує заходи, створені задля підвищення кваліфікації членів команди розробників (глава 22) |

У лістингу 4.1 показаний процес планування створення у вигляді псевдокода. Тут наголошено на тому, що планування – це ітераційний процес. Оскільки у процесі виконання проекту постійно надходить нова інформація, план має регулярно переглядатися. Важливими чинниками, які мають враховуватися під час розробки плану, є фінансові та ділові зобов'язання організації. Якщо вони змінюються, ці зміни також повинні знайти відображення у плані робіт.

Лістинг 4.1. Процес планування проекту

Визначення проектних обмежень

Початкова оцінка параметрів проекту

Визначення етапів виконання проекту та контрольних позначок

**while**поки проект не завершиться або не буде зупинено loop

Складання графіка робіт

Початок виконання робіт

Очікування закінчення чергового етапу робіт

Відстеження ходу виконання робіт

Перегляд оцінок параметрів проекту

Зміна графіка робіт

Перегляд проектних обмежень

**if**(виникла проблема) then

Перегляд технічних чи організаційних параметрів проекту

**end if**

**end loop**

Процес планування починається з визначення проектних обмежень (тимчасові обмеження, можливості готівкового персоналу, бюджетні обмеження тощо). Ці обмеження повинні визначатись паралельно з оцінюванням проектних параметрів, таких як структура та розмір проекту, а також розподілом функцій серед виконавців. Потім визначаються етапи розробки та те, які результати (документація, прототипи, підсистеми або версії програмного продукту) мають бути отримані після цих етапів. Далі починається циклічна частина планування. Спочатку розробляється графік робіт з виконання проекту або надається дозвіл на продовження використання раніше створеного графіка. Після цього (зазвичай через 2-3 тижні) проводиться контроль виконання робіт та відзначаються розбіжності між реальним та плановим ходом робіт.

Далі, у міру надходження нової інформації про хід виконання проекту можливий перегляд початкових оцінок параметрів проекту. Це, своєю чергою, може призвести до зміни графіка робіт. Якщо в результаті цих змін порушуються терміни завершення проекту, мають бути переглянуті (і погоджені із замовником ПЗ) проектні обмеження.

Звичайно, більшість менеджерів проектів не думають, що реалізація їх проектів пройде гладко, без жодних проблем. Бажано описати можливі проблеми ще до того, як вони виявлять себе під час виконання проекту. Тому краще складати "песимістичні" графіки робіт, ніж "оптимістичні". Але, звичайно, неможливо побудувати план, що враховує всі, в тому числі випадкові, проблеми та затримки виконання проекту, тому й виникає необхідність періодичного перегляду проектних обмежень та етапів створення програмного продукту.

**4.2.1. План проекту**

План проекту має чітко показати ресурси, необхідні для реалізації проекту, поділ робіт на етапи та тимчасовий графік виконання цих етапів. У деяких організаціях план проекту складається як єдиний документ, що містить усі види планів, описаних вище. В інших випадках план проекту описує лише технологічний процес створення ПЗ. У такому плані обов'язково є посилання на плани інших видів, але вони розробляються окремо від плану проекту.

План, структуру якого я подаю нижче, належить саме до останнього типу планів. Деталізація планів проектів дуже різниться залежно від типу програмного продукту, що розробляється, і організації-розробника. Але у будь-якому разі більшість планів містять такі розділи.

1. *Введення.*Короткий опис цілей проекту та проектних обмежень (бюджетних, тимчасових тощо), які є важливими для управління проектом.

2. *Організація виконання проекту.*Опис способу підбору команди розробників та розподіл обов'язків між членами команди.

3. *Аналіз ризиків.*Опис можливих проектних ризиків, ймовірності їхнього прояву та стратегій, спрямованих на їх зменшення. Тему управління ризиками розглянуто у розділі 4.4.

4. *Апаратні та програмні ресурси, необхідні для реалізації проекту.*Перелік апаратних засобів та програмного забезпечення, необхідного для розробки програмного продукту. Якщо апаратні засоби потрібно закуповувати, наводиться їх вартість разом із графіком закупівлі та постачання.

5. *Розбиття робіт на етапи.*Процес реалізації проекту розбивається на окремі процеси, визначаються етапи виконання проекту, наводиться опис результатів ("виходів") кожного етапу та контрольні позначки. Ця тема представлена ​​у розділі 4.2.2.

6. *Графік роботи.*У цьому графіку відображаються залежності між окремими процесами (етапами) розробки програмного забезпечення, оцінки часу їх виконання та розподіл членів команди розробників за окремими етапами.

7. *Механізми моніторингу та контролю за ходом виконання проекту.*Описуються звіти, що надаються менеджером про хід виконання робіт, терміни їх надання, а також механізми моніторингу всього проекту.

План має регулярно переглядатися у процесі реалізації проекту. Одні частини плану, наприклад графік робіт, часто змінюються, інші більш стабільні. Для внесення змін до плану потрібна спеціальна організація документопотоку, що дозволяє відстежувати ці зміни.

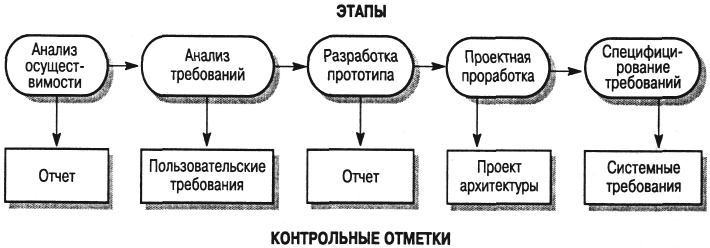
**4.2.2. Контрольні позначки етапів робіт**

Менеджеру для організації процесу створення ПЗ та управління їм потрібна інформація. Оскільки саме програмне забезпечення невловиме, ця управлінська інформація може бути отримана лише у вигляді документів, що відображають виконання чергового етапу розробки програмного продукту. Без цієї інформації не можна судити про ступінь готовності продукту, неможливо оцінити вироблені витрати або змінити графік робіт.

При плануванні процесу визначаються контрольні позначки – віхи, які відзначають закінчення певного етапу работ. Для кожної контрольної позначки створюється звіт, який надається керівництву проекту. Ці звіти не мають бути більшими об'ємними документами; вони мають підбивати короткі підсумки закінчення окремого логічно завершеного етапу проекту. Етапом не може бути, наприклад, "Написання 80% коду програм", оскільки неможливо перевірити завершення такого "етапу": крім того, подібна інформація практично не є корисною для управління, оскільки тут не відображається зв'язок цього "етапу" з іншими етапами створення ПЗ.

Зазвичай після завершення основних великих етапів, як-от розробка специфікації, проектування тощо, замовнику ПЗ надаються результати їх виконання, звані контрольні проектні елементи. Це може бути документація, прототип програмного продукту, закінчені підсистеми і т.д. Контрольні проектні елементи, що надаються замовнику, можуть збігатися з контрольними відмітками (точніше, з результатами виконання будь-якого етапу). Але зворотне твердження не так. Контрольні позначки – це внутрішні проектні результати, які використовуються контролю за ходом виконання проекту, і вони, зазвичай, не надаються замовнику ПЗ.

Для визначення контрольних позначок весь процес створення ПЗ має бути розбитий на окремі етапи із зазначеним "виходом" (результатом) кожного етапу. Наприклад, на рис. 4.1 показано етапи розробки специфікації вимог у разі, коли для її перевірки використовується прототип системи, а також подано вихідні результати (контрольні позначки) кожного етапу. Тут контрольними проектними елементами є вимоги та специфікація вимог.



*Мал. 4.1. Етапи процесу розробки специфікації*

### 4.3. Графік робіт

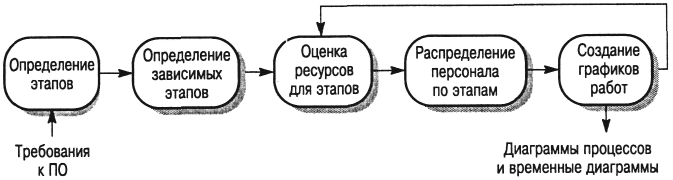
Складання графіка – одна з найвідповідальніших робіт, які виконує менеджер проекту. Тут менеджер оцінює тривалість проекту, визначає ресурси, необхідних реалізації окремих етапів робіт, і представляє їх (етапи) як узгодженої послідовності. Якщо цей проект подібний до раніше реалізованого, то графік робіт останнього проекту можна взяти за основу для цього проекту. Але потім слід врахувати, що на окремих етапах нового проекту можуть використовуватися методи та підходи, відмінні від використаних раніше.

Якщо проект є інноваційним, початкові оцінки тривалості та необхідних ресурсів, напевно, будуть надто оптимістичними, навіть якщо менеджер спробує передбачити всі можливі несподіванки. З цього погляду проекти створення не відрізняються від великих інноваційних технічних проектів. Нові аеропорти, мости і навіть нові автомобілі, як правило, з'являються пізніше спочатку оголошених термінів їх здачі або надходження на ринок, чому причиною є проблеми і труднощі, що несподівано виникли. Саме тому графіки робіт необхідно постійно оновлювати у міру надходження нової інформації про хід виконання проекту.

У процесі складання графіка (рис. 4.2) весь масив робіт, необхідні реалізації проекту, розбивається окремі етапи і оцінюється час, необхідний виконання кожного этапа. Зазвичай багато етапів виконуються паралельно. Графік робіт повинен передбачати це та розподіляти виробничі ресурси між ними найбільш оптимальним чином. Нестача ресурсів для виконання будь-якого критичного етапу – найчастіша причина затримки виконання всього проекту.

Тривалість етапів зазвичай має бути не менше тижня. Якщо вона буде меншою, то виявиться нижчою за точність тимчасових оцінок етапів, що може призвести до частого перегляду графіка робіт. Також доцільно (в аспекті управління проектом) встановити максимальну тривалість етапів, що не перевищує 8 чи 10 тижнів. Якщо є етапи, які мають більшу тривалість, їх слід розбити на етапи меншої тривалості.

При розрахунку тривалості етапів менеджер повинен враховувати, що виконання будь-якого етапу не обійдеться без великих чи дрібних проблем і затримок. Розробники можуть припускатися помилок або затримувати свою роботу, техніка може вийти з ладу або апаратні або програмні засоби підтримки процесу розробки можуть надійти із запізненням. Якщо проект інноваційний та технічно складний, це стає додатковим фактором появи непередбачених проблем та збільшення тривалості реалізації проекту порівняно з початковими оцінками.



*Мал. 4.2. Процес складання графіка робіт*

Крім тимчасових витрат, менеджер має розрахувати інші ресурси, необхідні успішного виконання кожного етапу. Особливий вид ресурсів – це команда розробників, залучена до виконання проекту. Іншими видами ресурсів можуть бути необхідний вільний дисковий простір на сервері, час використання будь-якого спеціального обладнання та бюджетні кошти на витрати на відрядження персоналу, що працює над проектом. Більш детально види та методи оцінювання необхідних ресурсів розглядаються у розділі 23.

Існує гарне емпіричне правило: оцінювати тимчасові витрати так, начебто нічого непередбачуваного та "поганого" не може статися, потім збільшити ці оцінки для врахування можливих проблем. Можливі, але важко прогнозовані проблеми суттєво залежать від типу та параметрів проекту, а також від кваліфікації та досвіду членів команди розробників. Так як це правило емпіричне, дозволю дати пораду, засновану на моєму досвіді. До вихідних розрахункових оцінок я завжди додаю 30% на можливі проблеми, а потім ще 20%, щоб бути готовим до того, що я не можу передбачити.

Графік робіт за проектом зазвичай представляється у вигляді набору діаграм та графіків, що показують розбиття проектних робіт на етапи, залежності між етапами та розподіл розробників за етапами. Ці діаграми розглядаються у наступному розділі. Зазначу, що зараз існує багато різних програмних засобів підтримки управління проектами, наприклад Microsoft Project.

**4.3.1. Тимчасові та мережеві діаграми**

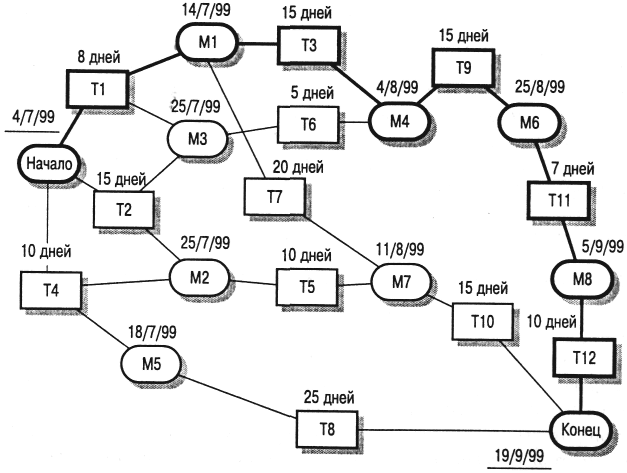
Тимчасові та мережеві діаграми корисні для представлення графіка робіт. Тимчасова діаграма показує час початку та закінчення кожного етапу та його тривалість. Мережева діаграма відображає залежність між різними етапами проекту. Ці діаграми можна створити автоматично за допомогою програмних засобів підтримки керування на основі інформації, закладеної в базі даних проекту.

Розглянемо етапи якогось проекту, подані у табл. 4.2 з якої, зокрема, видно, що етап Т3 залежить від етапу Т1. Це означає, що етап Т1 повинен завершитися, перш ніж почнеться етап Т3. Наприклад, етапі Т1 проводиться компонентний аналіз створюваного програмного продукту, але в етапі Т3 – проектування системи.

На основі наведених значень тривалості етапів та залежності між ними будується мережевий графік послідовності етапів (рис. 4.3). У цьому графіку видно, які роботи можуть виконуватися паралельно, які повинні виконуватися послідовно друг за одним. Етапи позначені прямокутниками. Контрольні позначки та контрольні проектні елементи показані у вигляді овалів та позначені (як і в табл. 4.2) літерою М з відповідним номером. Дати на цій діаграмі відповідають початку виконання етапів. Мережеву діаграму слід читати зліва направо та зверху вниз.

**Таблиця 4.2. Етапи проекту**

|  |  |  |
| --- | --- | --- |
| **Етап** | **Тривалість (дні)** | **Залежність** |
| Т1  Т2  Т3  Т4  Т5  Т6  Т7  Т8  Т9  Т10  Т11  Т12 | 8  15  15  10  10  5  20  25  15  15  7  10 | Т1(М1)  Т2, Т4 (М2)  Т1, Т2 (МОЗ)  Т1(М1)  Т4 (М5)  Т3, Т6 (М4)  Т5, Т7 (М7)  Т9 (Мб)  Т11(М8) |



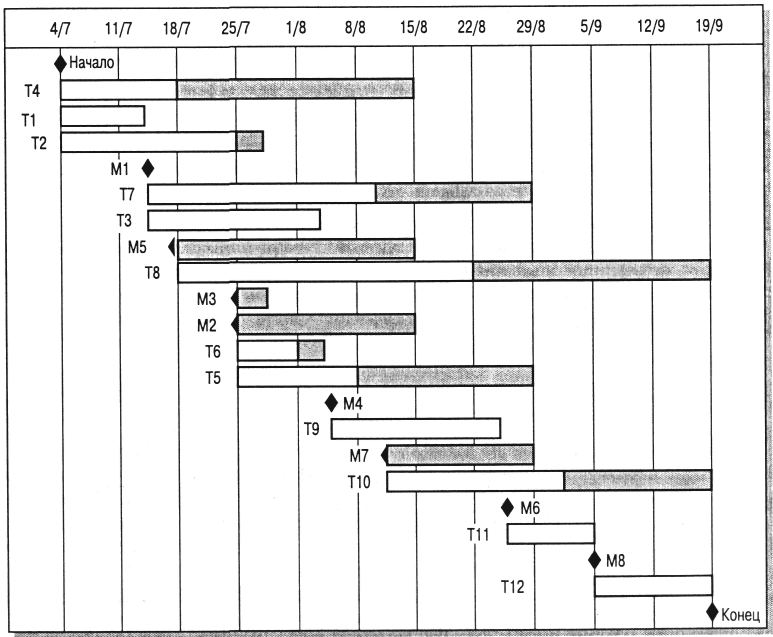
*Мал. 4.3. Мережева діаграма етапів*

Якщо для створення мережної діаграми використовуються програмні засоби підтримки управління проектом, то кожен етап повинен закінчуватися контрольною відміткою. Черговий етап може розпочатися лише тоді, коли буде отримано контрольну позначку (яка може залежати від кількох попередніх етапів). Тому у третьому стовпці табл. 4.2 наведено контрольні позначки; вони будуть досягнуті лише тоді, коли буде завершено етап, у рядку якого вміщено відповідну контрольну позначку.

Будь-який етап не може початися, доки не виконані всі етапи на всіх шляхах, що ведуть від початку проекту до цього етапу. Наприклад, етап Т9 не може початися, доки не будуть завершені етапи ТЗ і Т6. Зазначимо, що у разі досягнення контрольної позначки М4 свідчить, що це етапи завершено.

Мінімальний час виконання всього проекту можна розрахувати, підсумувавши в мережній діаграмі тривалості етапів на найдовшому шляху від початку проекту до його закінчення (це так званий критичний шлях). У разі тривалість проекту становить 11 тижнів чи 55 робочих днів. На рис. 4.3 критичний шлях показаний товстішими лініями, ніж інші шляхи. Таким чином, загальна тривалість реалізації проекту залежить від етапів робіт на критичному шляху. Будь-яка затримка на завершення будь-якого етапу на критичному шляху призведе до затримки всього проекту.

*\* Довжина колії тут вимірюється не кількістю етапів на дорозі, а сумарною тривалістю цих етапів.*- Прим. ред.



*Мал. 4.4. Тимчасова діаграма тривалості етапів*

Затримка у завершенні етапів, які входять у критичний шлях, впливає тривалість всього проекту до того часу, поки сумарна тривалість цих етапів (з урахуванням затримок) якомусь шляху не перевищить тривалості робіт на критичному шляху. Наприклад, затримка етапу Т8 на строк, менший за 20 днів, ніяк не впливає на загальну тривалість проекту. На рис. 4.4 представлена ​​тимчасова діаграма, де показані можливі затримки кожному етапі.

Мережева діаграма дозволяє побачити залежно етапів значимість тієї чи іншої етапу реалізації всього проекту. Увага до етапів критичного шляху часто дозволяє знайти способи їх зміни для того, щоб скоротити тривалість всього проекту. Менеджери використовують мережну діаграму для розподілу робіт.

На рис. 4.4 показано інше уявлення графіка робіт. Це часова діаграма (іноді звана на ім'я її винахідника діаграмою Гантта) може бути побудована програмними засобами підтримки процесу управління. Вона показує тривалість виконання кожного етапу та можливі їх затримки (показані затіненими прямокутниками), а також дати початку та закінчення кожного етапу. Етапи критичного шляху немає затінених прямокутників; це означає, що затримка із завершенням даних етапів призведе до збільшення тривалості всього проекту.

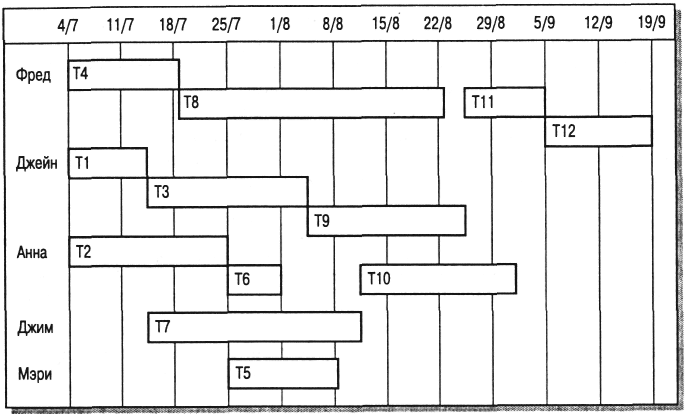
Подібно до розподілу часу виконання етапів, менеджер повинен розрахувати розподіл ресурсів по етапах, зокрема призначити виконавців на кожен етап. У табл. 4.3 наведено розподіл розробників кожний етап, поданий на рис. 4.4.

Таблиця 4.3. Розподіл виконавців за етапами

|  |  |
| --- | --- |
| **Етап** | Виконавець |
| Т1  Т2  Т3  Т4  Т5  Т6  Т7  Т8  Т9  Т10  Т11  Т12 | Джейн  Анна  Джейн  Фред  Мері  Анна  Джим  Фред  Джейн  Анна  Фред  Фред |

Наведена таблиця може бути використана програмними засобами підтримки процесу управління для побудови тимчасової діаграми зайнятості працівників на певних етапах роботи (рис. 4.5). Персонал не зайнятий у роботі над проектом постійно його реалізації. Протягом періоду незайнятості співробітники може бути у відпустці, працювати з іншими проектами, проходити навчання тощо.

У великих організаціях зазвичай працює багато фахівців, які задіяні у проекті в міру необхідності. Звичайно такий підхід може створити певні проблеми для менеджерів проектів. Наприклад, якщо фахівець зайнятий у проекті, який затримується, це може створити прямі складнощі для інших проектів, де він також має брати участь.



*Мал. 4.5. Тимчасова діаграма розподілу працівників за етапами*

Початковий графік робіт неминуче містить якісь помилки чи недоробки. У міру реалізації проекту розраховані оцінки тривалості виконання етапів робіт мають порівнюватись із реальними термінами виконання цих етапів. Результати порівняння повинні використовуватися як основа для перегляду графіка робіт ще не реалізованих етапів проекту, зокрема для того, щоб спробувати зменшити тривалість етапів критичного шляху.

### 4.4. Управління ризиками

Важливою частиною роботи менеджера проекту є оцінка ризиків, які можуть вплинути на графік робіт або на якість програмного продукту, що створюється, і розробка заходів щодо запобігання ризикам. Результати аналізу ризиків мають бути відображені у плані проекту. Визначення ризиків та розробка заходів щодо зменшення їх впливу на хід виконання проекту називається управлінням ризиками [149, 266, 11\*, 23\*].

Спрощено ризик можна розуміти як ймовірність прояву будь-яких несприятливих обставин, що негативно впливають на реалізацію проекту. Ризики можуть загрожувати проекту загалом, створюваному програмним продуктом чи організації-разработчику. Можна виділити три типи ризиків.

1. *Ризики для проекту,*які впливають графік робіт чи ресурси, необхідні виконання проекту.

2. *Ризики для продукту, що розробляється.,*що впливають на якість або продуктивність програмного продукту, що розробляється.

3. *Бізнес-ризики,*що стосуються організації-розробника або постачальників.

Звісно, ​​ці типи ризиків можуть перетинатися. Наприклад, якщо досвідчений програміст залишає проект, це буде ризиком для проекту (оскільки затримується термін здачі готового продукту), ризиком для продукту (оскільки новий програміст, який замінив пішов, може виявитися не надто досвідченим і зробити помилки в програмі) і бізнес-ризиком ( оскільки затримка цього проекту може негативно вплинути на майбутні ділові контакти між замовником та організацією-розробником).

Конкретні типи ризиків, які можуть вплинути цей проект, залежить від виду створюваного програмного продукту і зажадав від організаційного оточення, де реалізується програмний проект. Разом з тим, багато типів ризиків здатні вплинути на будь-які програмні проекти, ці ризики наведені в табл. 4.4.

**Таблиця 4.4. Можливі ризики програмних проектів**

|  |  |  |
| --- | --- | --- |
| **Ризик** | **Тип ризику** | **Опис ризику** |
| Плинність розробників  Зміна в управлінні організацією  Неготовність апаратних засобів  Зміна вимог  Затримка у розробці специфікації  Недооцінка розміру системи, що розробляється  Недостатня ефективність CASE-засобів  Зміни у технології розробки ПЗ  Поява конкуруючого програмного продукту | Ризик для проекту  Ризик для проекту  Ризик для проекту  Ризик для проекту та продукту, що розробляється  Ризик для проекту та продукту, що розробляється  Ризик для проекту та продукту, що розробляється  Ризик для продукту, що розробляється  Бізнес-ризик  Бізнес-ризик | Досвідчені розробники залишають проект до його завершення  Організація змінює свої пріоритети в управлінні проектом  Апаратні засоби, необхідні для проекту, не надійшли вчасно або не готові до експлуатації  Поява великої кількості непередбачених змін у вимогах до ПЗ, що розробляється  Специфікації основних інтерфейсів підсистем не надійшли до розробників відповідно до графіка робіт  Розмір системи значно перевищив початкову оцінку  CASE-кошти, призначені для підтримки проекту, виявилися менш ефективними, ніж очікувалося  Основні технології побудови програмної системи замінюються на нові  На ринку програмних продуктів до закінчення проекту з'явилася конкуруюча програмна система |

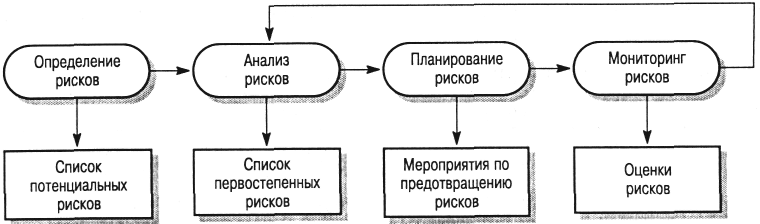
Схема процесу управління ризиками показано на рис. 4.6. Цей процес складається із чотирьох стадій.

1. *Визначення ризиків.*Визначаються можливі ризики для проекту, для продукту, що розробляється, і бізнес-ризики.

2. *Аналіз ризиків.*Оцінюється ймовірність та послідовність появи ризикових ситуацій.

3. *Планування ризиків.*Плануються заходи щодо запобігання ризикам або мінімізації їх впливу на проект.

4. *Моніторинг ризиків*Постійне оцінювання ймовірностей ризиків та виконання заходів щодо пом'якшення наслідків прояву ризикових ситуацій.



*Мал. 4.6. Процес управління ризиками*

Процес управління ризиками, як та інші процеси планування, є ітераційним, виконуваним протягом усього терміну реалізації проекту. Спочатку розробляються плани управління ризиками, потім постійно відстежується ситуація довкола реалізації проекту. При надходженні нової інформації про можливі ризики наново проводиться аналіз ризиків і першочергова увага приділяється новим ризикам. У міру надходження нової інформації також змінюються плани заходів щодо запобігання та пом'якшення ризиків.

Результати процесу управління ризиками документуються як планів управління ризиками. Вони повинні включати опис можливих проектних ризиків, їх аналіз та перелік заходів, необхідних для управління ризиками.

**4.4.1. Визначення ризиків**

Визначення ризиків – перша стадія процесу управління ризиками. На цій стадії описуються ризики, які можуть проявитися під час реалізації проекту. У принципі, на цій стадії не повинна оцінюватися ймовірність і значущість ризиків, але на практиці малоймовірні ризики з незначними наслідками зазвичай відкидаються одразу.

Визначення ризиків може виконуватися в режимі командної роботи з використанням підходу "мозковий штурм" або ґрунтуватися на досвіді менеджера. При визначенні ризиків може допомогти наведений нижче перелік можливих категорій ризиків.

1. *Технологічні ризики*Походять із програмних та апаратних технологій, на основі яких розробляється система.

2. *Ризики, пов'язані з персоналом.*Пов'язані із членами команди розробників.

3. *Організаційні ризики.*Випливають із організаційного оточення, в якому виконується проект.

4. *Інструментальні ризики.*Пов'язані з CASE-засобами та іншими засобами підтримки процесу створення ПЗ.

5. *Ризики, пов'язані із системними вимогами.*Виявляються при зміні вимог, що пред'являються до системи, що розробляється.

6. *Ризики оцінювання.*Пов'язані з оцінюванням характеристик програмної системи та ресурсів, необхідні реалізації проекту.

У табл. 4.5 наведено деякі приклади, що стосуються кожної з описаних категорій ризиків. Результатом етапу визначення ризиків буде довгий список можливих ризиків, які можуть вплинути на програмний продукт, проект або організацію-розробника, що розробляється.

**Таблиця 4.5. Категорії ризиків**

|  |  |
| --- | --- |
| **Категорія ризиків** | **Приклади ризиків** |
| Технологічні ризики  Ризики, пов'язані з персоналом  Організаційні ризики  Інструментальні ризики  Ризики, пов'язані із системними вимогами | База даних, яка використовується у програмній системі, не забезпечує обробку очікуваного обсягу транзакцій.  Програмні компоненти, які використовуються повторно, мають дефекти, що обмежують їх функціональні можливості  Неможливо підібрати працівників із необхідним професійним рівнем.  Ведучий розробник захворів у найкритичніший час.  Неможливо організувати необхідне навчання персоналу  В організації, що виконує розробку програмного забезпечення, відбулася реорганізація, внаслідок чого змінилися пріоритети в управлінні проектом.  Фінансові труднощі в організації призвели до зменшення бюджету проекту  Програмний код, що генерується CASE-засобами, не є ефективним.  CASE-засоби неможливо інтегрувати з іншими засобами підтримки проекту  Зміни вимог призводять до значних повторних робіт із проектування системи.  Початкове нечітке формулювання вимог користувача призвело до значних змін системних вимог, що проявилися на пізніх стадіях розробки проекту. |
| Ризики оцінювання | Недооцінки часу виконання проекту.  Швидкість виявлення дефектів у системі нижча за раніше заплановану.  Розмір системи значно перевищує спочатку розрахований |

**4.4.2. Аналіз ризиків**

При аналізі для кожного певного ризику підраховується ймовірність його прояву та збитки, які він може завдати. Не існує простих методів виконання аналізу ризиків – значною мірою він ґрунтується на думці та досвіді менеджера. Не претендуючи на виняткову точність, можна навести таку шкалу ймовірностей ризиків та їх наслідків.

1. Можливість ризику вважається дуже низькою, якщо вона має значення менше 10%; низькою, якщо її значення від 10 до 25%; середньої при значеннях від 25 до 50%; високою, якщо значення коливається від 50 до 75%; дуже високою при значеннях понад 75%.

2. Можливі збитки від ризикових ситуацій можна поділити на катастрофічні, серйозні, терпимі і незначні.

Результати аналізу ризиків мають бути представлені у вигляді таблиці ризиків, упорядкованих за ступенем можливої ​​шкоди. У табл. 4.6 наведено впорядкований перелік ризиків, описаних у табл4.5; там же вказано ймовірність цих ризиків. Тут ймовірності ризиків та ступінь збитків від них вказані довільно. На практиці для їх визначення необхідна докладна інформація про проект, технологію створення ПЗ, команду розробників і про саму організацію.

**Таблиця 4.6. Список ризиків після проведення їх аналізу**

|  |  |  |
| --- | --- | --- |
| **Ризик** | **Ймовірність** | **Ступінь шкоди** |
| Фінансові труднощі в організації призвели до зменшення бюджету проекту | Низька | Катастрофічна |
| Неможливо підібрати працівників із потрібним професійним рівнем | Висока | Катастрофічна |
| Ведучий розробник захворів у найкритичніший час | Середня | Серйозна |
| Програмні компоненти, які використовуються повторно, мають дефекти, що обмежують їх функціональні можливості | Середня | Серйозна |
| Зміни вимог призводять до значних повторних робіт із проектування системи | Середня | Серйозна |
| В організації, що виконує розробку ПЗ, відбулася реорганізація, внаслідок чого змінилися пріоритети в управлінні проектом | Висока | Серйозна |
| База даних, що використовується в програмній системі, не забезпечує обробку очікуваного обсягу транзакцій | Середня | Серйозна |
| Недооцінки часу виконання проекту | Висока | Серйозна |
| CASE-засоби неможливо інтегрувати з іншими засобами підтримки проекту | Висока | Терпима |
| Початкове нечітке формулювання вимог користувача призвело до значних змін системних вимог, що проявилися на пізніх стадіях розробки проекту. | Середня | Терпима |
| Неможливо організувати необхідне навчання персоналу | Середня | Терпима |
| Швидкість виявлення дефектів у системі нижче запланованої раніше | Середня | Терпима |
| Розмір системи значно перевищує спочатку розрахований | Висока | Терпима |
| Програмний код, що генерується CASE-засобами, неефективний | Середня | Незначна |

Звичайно, як ймовірність ризиків, так і можливі збитки від них повинні переглядатися при надходженні додаткової інформації про ці ризики та в міру реалізації заходів з управління ними. Тому такі таблиці ризиків мають перероблятися кожної ітерації процесу управління ризиками.

Після аналізу ризиків визначаються найбільш значущі ризики, які потім відстежуються протягом усього терміну виконання проекту. Визначення цих значних ризиків залежить від їх ймовірностей та можливої ​​шкоди. У загальному випадку завжди відслідковуються ризики з катастрофічними наслідками, а також ризики з серйозними збитками, значення ймовірності яких вище середнього.

У статті [47] рекомендується визначити та відслідковувати "10 верхніх" ризиків, але я думаю, що це необґрунтована рекомендація. Кількість ризиків, які слід відстежувати, залежить від конкретного проекту. Це може бути п'ять ризиків, а може п'ятнадцять. Але, звичайно, кількість ризиків, за якими проводиться моніторинг, має бути доступною для огляду. Велика кількість відстежуваних ризиків вимагатиме величезної кількості інформації, що збирається. Зі списку ризиків, поданих у табл. 4.6, для моніторингу слід відібрати всі вісім ризиків, які можуть призвести до катастрофічних та серйозних наслідків.

**4.4.3. Планування ризиків**

Планування полягає у визначенні стратегії управління кожним значним ризиком, відібраним для моніторингу після аналізу ризиків. Тут також не існує загальноприйнятих підходів для розробки таких стратегій – багато що ґрунтується на "чутті" та досвіді менеджера проекту. У табл. 4.7 показано можливі стратегії управління основними ризиками, наведеними у табл. 4.6.

**Таблиця 4.7. Стратегії управління ризиками**

|  |  |
| --- | --- |
| **Ризик** | Стратегія |
| Фінансові проблеми організації | Підготувати короткий документ для керівництва організації, що показує важливість цього проекту для досягнення фінансових цілей організації |
| Проблеми некваліфікованого персоналу | Попередити замовника про потенційні труднощі та можливу затримку проекту, розглянути питання про купівлю компонентів системи |
| Хвороби персоналу | Реорганізувати роботу команди розробників таким чином, щоб обов'язки та робота членів команди перекривали один одного, внаслідок цього розробники знатимуть та розумітимуть завдання, які виконують інші співробітники. |
| Дефектні системні компоненти | Замінити потенційно дефектні системні компоненти покупними компонентами, що гарантують якість роботи |
| Зміни вимог | Спробувати визначити вимоги, що найімовірніше схильні до змін; у структурі системи не відображатиме детальну інформацію |
| Реорганізація компанії-розробника | Підготувати короткий документ для керівництва компанії, що показує важливість цього проекту для досягнення фінансових цілей компанії |
| Недостатня продуктивність бази даних  Недооцінки часу виконання проекту | Розглянути можливість купівлі більш продуктивної бази даних  Розглянути питання купівлі системних компонентів, дослідити можливість використання генератора програмного коду |

Існує три категорії стратегій управління ризиками.

1. *Стратегії запобігання ризикам.*Відповідно до цих стратегій слід проводити заходи, що знижують ймовірність прояву ризиків. Прикладом може бути стратегія виключення потенційно дефектних компонентів, описана в табл. 4.7.

2. *Мінімізаційні стратегії.*Спрямовані на зменшення можливої ​​шкоди від ризиків. Прикладом є стратегія зменшення шкоди від хвороби членів команди розробників (див. табл. 4.7).

3. *Планування "аварійних" ситуацій.*Відповідно до цих стратегій необхідно мати план заходів, які слід виконати у разі прояву ризикової ситуації. У табл. 4.7 це стратегія поведінки у разі виникнення фінансових проблем у організації-розробника.

**4.4.4. Моніторинг ризиків**

Моніторинг ризиків полягає в регулярному перерахуванні ймовірностей ризиків та збитків, які вони можуть завдати. Для цього необхідно постійно відстежувати фактори, що впливають на ймовірність ризиків та можливі збитки. Ці чинники залежить від типів ризику. У табл. 4.8 наведено ознаки, що допомагають визначити тип ризику.

**Таблиця 4.8. Ознаки ризиків**

|  |  |
| --- | --- |
| **Тип ризику** | **Ознаки** |
| Технологічні ризики | Затримки у постачанні обладнання або програмних засобів підтримки процесу створення ПЗ, численні документовані технологічні проблеми |
| Ризики, пов'язані з персоналом | Низький моральний стан персоналу, натягнуті відносини між членами команди розробників, низька якість виконаної роботи |
| Організаційні ризики | Розмови серед персоналу про пасивність та недостатню компетентність вищого керівництва організації |
| Інструментальні ризики | Небажання розробників використовувати програмні засоби підтримки, несхвальні відгуки про CASE-засоби, запити на потужніші інструментальні засоби |
| Ризики, пов'язані із системними вимогами | Необхідність перегляду багатьох системних вимог, невдоволення замовника ПЗ |
| Ризики оцінювання | Зміни графіка робіт, численні звіти щодо порушення графіка робіт |

Моніторинг ризиків повинен бути безперервним процесом, який відстежує хід виконання заходів з управління ризиками, при цьому кожен основний ризик має розглядатися окремо.

**КЛЮЧОВІ ПОНЯТТЯ**

• Ознакою хорошого управління програмним проектом є виконання проекту відповідно до графіка робіт та в рамках запланованого бюджету.

• Управління програмними проектами відрізняється від управління іншими технічними проектами, оскільки програмне забезпечення нематеріальне. Досвіду, накопиченого при реалізації більш ранніх проектів, може виявитися недостатньо для вироблення стратегії управління новими та складними програмними проектами.

• Менеджер програмного проекту виконує безліч різнотипних робіт. Серед них основними є планування проекту, оцінювання тимчасових та інших ресурсів, необхідні реалізації проекту, і навіть складання графіка робіт. Планування проекту та оцінювання необхідних ресурсів – це ітераційні види діяльності, які продовжуються протягом усього терміну виконання проекту. У міру надходження додаткової інформації про хід виконання проекту плани та графіки можуть переглядатись.

• Контрольні позначки – це прогнозовані "виходи" етапів реалізації проектів, які разом із звітом про виконання етапу передаються керівництву проектом. Контрольні проектні елементи – контрольні позначки, які надаються замовнику програмної системи.

• Складання графіка робіт полягає у створенні різних графічних уявлень окремих частин плану проекту. Сюди відносяться мережеві діаграми етапів, що показують взаємозалежність етапів робіт, та часові діаграми тривалості етапів.

• Для будь-якого проекту мають бути визначені основні ризики, а також ймовірність їх здійснення та можливі збитки від них. Для найбільш ймовірних ризиків, наслідки від яких більш ніж серйозні, розробляються заходи щодо запобігання або зниження можливої ​​шкоди.

Вправи

**4.1.** Поясніть, чому нематеріальність програмних систем породжує особливі проблеми у процесі керування програмними проектами.

**4.2.** Поясніть, чому хороші програмісти не завжди можуть бути хорошими менеджера проектів. Для побудови пояснення може допомогти список робіт, які виконує менеджер проектів (див. розділ 4.1).

**4.3.** Поясніть, чому процес планування проекту є ітераційним і чому план повинен постійно переглядатись протягом усього терміну виконання проекту.

**4.4.** Опишіть коротко кожен розділ плану виконання програмного проекту.

**4.5.** У чому принципова різниця між контрольною відміткою та контрольним програмним елементом?

**4.6.** У табл. 4.9 показані етапи якогось проекту, їх тривалість та взаємозалежність між ними. Намалюйте мережну та часову діаграми етапів робіт для даного проекту.

**Таблиця 4.9. Етапи проекту**

|  |  |  |
| --- | --- | --- |
| **Етап** | **Тривалість (дні)** | **Залежність** |
| Т1  Т2  Т3  Т4  Т5  Т6  Т7  Т8  Т9  Т10  Т11  Т12  Т13  Т14  Т15  Т16 | 10  15  10  20  10  15  20  35  15  5  10  20  35  10  20  10 | Т1  Т1, Т2  Т3, Т4  Т3  Т7  Т6  Т5, Т9  Т9  Т10  Т3, Т4  Т8, Т9  Т12, Т14  Т15 |

**4.7.** У табл. 4.2 наведено тривалість етапів деякого проекту. Припустимо, що через деякі серйозні причини етап Т5 був виконаний за 40 днів замість запланованих 10 днів. Переробіть мережну діаграму етапів та визначте новий критичний шлях. Намалюйте нову часову діаграму.

**4.8.** На додаток до ризиків, наведених у табл. 4.5, визначте ще шість ризиків, можливих під час реалізації проекту.

**4.9.** Менеджер проекту попереджає про можливу затримку виконання робіт, якої можна уникнути лише за рахунок безкоштовних понаднормових робіт команди розробників. Усі члени команди мають сім'ї, які потребують певної частки уваги. Обговоріть можливість відхилення пропозиції менеджера про безкоштовні понаднормові роботи або згоду надати перевагу інтересам організації сімейних інтересів. Які аргументи найвагоміші у цій дискусії?

**4.10.** Як досвідченому програмісту вам запропонували очолити управління проектом, але ви відчуваєте, що більше користі можете принести в якості технічного фахівця, а не менеджера проекту.Обговоріть можливості прийняття чи відхилення пропозиції очолити програмний проект.