# Практична робота №3. Прототипування програмних системМета: Здобути навички використання прототипування в процесі розробки програмного забезпечення, і описати різні підходи до розробки прототипів.

# Теоретичні відомості

Замовникам програмного забезпечення і кінцевим користувачам зазвичай складно чітко сформулювати вимоги до розроблювальної програмної системи. Важко передбачити, як система буде впливати на трудовий процес, як вона буде взаємодіяти з іншими системами і які операції, виконувані користувачами, необхідно автоматизувати. Ретельний аналіз вимог допомагає зменшити невизначеність щодо того, що система повинна робити. Однак реально перевірити вимоги, перш ніж їх затвердити, практично неможливо. У цій ситуації може допомогти прототип системи.

Прототип є початковою версією програмної системи, яка використовується для демонстрації концепцій, закладених у системі, перевірки варіантів вимог, а також пошуку проблем, які можуть виникнути як у ході розробки, так і при експлуатації системи, і можливих варіантів їх розв'язку.

Прототип ПЗ допомагає на двох етапах процесу розробки системних вимог.

1. ***Постановка вимог.*** Користувачі можуть експериментувати із системними прототипами, що дозволяє їм перевіряти, як буде працювати система. Користувачі одержують нові ідеї для постановки вимог, можуть визначити сильні і слабкі сторони ПЗ. У результаті можуть сформуватися нові вимоги.
2. ***Перевірка вимог.*** Прототип дозволяє виявити помилки і недогляд в раніше прийнятих вимогах. Наприклад, системні функції, визначені у вимогах, можуть бути корисними і потрібними (з погляду користувача). Однак у процесі застосування цих функцій разом з іншими функціями користувачі можуть змінити первісну думка про них. У результаті вимоги до системи зміняться, відбиваючи змінене розуміння користувачами системних функцій.

Прототипування можна використовувати при аналізі ризиків і на початковому етапі розробки планів керування програмним проектом. Основною небезпекою при розробці ПЗ є помилки і недогляд у вимогах. Витрати на усунення помилок у вимогах на більш пізніх стадіях процесу розробки можуть бути дуже високими. Експерименти показують, що прототипування зменшує число проблем, пов'язаних з розробкою вимог. Крім того, прототипування зменшує загальну вартість розробки системи. Із цих причин воно часто використовується в процесі розробки вимог.

Однак відмінність між прототипуванням, як окремим етапом процесу розробки ПЗ, і розробкою основної програмної системи неочевидно. У цей час багато систем розробляються з використанням еволюційного підходу, коли швидко створюється первісна версія системи, яка потім поступово змінюється до її остаточного варіанта. При цьому часто використовуються методи швидкої розробки додатків, які також можна використовувати при створенні прототипів.

Поряд з тим що прототипи допомагають формувати вимоги, вони мають і інші переваги.

1. Різне тлумачення вимог розробниками ПЗ і користувачами можна виявити при демонстрації діючого прототипу системи.
2. У процесі створення прототипу розробники можуть виявити неповні або неузгоджені вимоги.
3. Працюючи, хоча і обмежено, у вигляді прототипу, система може продемонструвати свої слабкі і сильні сторони.
4. Прототип може бути основою для написання специфікації високоякісної системи. Розробка прототипу зазвичай веде до поліпшення специфікації системи.

Діючий прототип може також використовуватися для інших цілей.

1. ***Навчання користувача.*** Прототип системи можна використовувати для навчання персоналу перед поставкою остаточного варіанта системи.
2. ***Тестування системи.*** Прототипи дозволяють “прокручувати ” тести. Той самий тест запускається на прототипі і на системі. Якщо виходять однакові результати, це означає, що тест не виявив дефектів у системі. Якщо результати відрізняються, то необхідно досліджувати причини відмінності, що дозволяє виявити можливі помилки в системі.

Ефективність застосування прототипів при розробці ПЗ полягає в наступному.

1. Поліпшуються експлуатаційні якості системи.
2. Система більше відповідає потребам користувачів.
3. Системна архітектура стає більш досконалою.
4. Супровід системи спрощується і стає більш зручним.
5. Скорочуються витрати на розробку системи.
6. Поліпшення експлуатаційних якостей системи і збільшення відповідності системи потребам користувача не вимагають збільшення загальної вартості розробки системи. Прототипування зазвичай підвищує вартість початкових етапів розробки ПЗ, але знижує витрати на більш пізніх етапах.

Як ми вже відзначали, кінцевим користувачам важко представити, як вони будуть використовувати нову систему ПЗ в повсякденній роботі. Якщо система більша і складна, то це неможливо зробити, перш ніж система буде створена і введена в експлуатацію.

Один зі способів подолання цих труднощів полягає у використанні еволюційного методу розробки систем. Це означає, що користувачеві надається незавершена система, яка потім змінюється і доповнюється доти, поки не стануть ясні всі вимоги користувача. У якості альтернативи можна побудувати “експериментальний” прототип, який допоможе проаналізувати і перевірити вимоги. Після цього створюється система.

Еволюційне прототипування починається з побудови відносно простої системи, яка реалізує найбільш важливі вимоги користувача. У міру виявлення нових вимог прототип змінюється і доповнюється. В остаточному підсумку він стає тою системою, яка потрібна. У цьому процесі не використовується детальна системна специфікація, у багатьох випадках немає навіть формального документа із системними вимогами. У цей час еволюційне прототипування є звичайною технологією розробки програмних систем, яка широко використовується при розробці Web-вузлів і додатків електронної комерції.

На противагу еволюційному підходу метод експериментального прототипування призначений для розробки і уточнення системної специфікації. Прототип створюється, оцінюється і модифікується. Дані оцінювання прототипу використовуються для подальшої деталізації специфікації. Коли системні вимоги сформовані, прототип більше не потрібний.

Існує відмінність між цілями еволюційного і експериментального прототипування.

* **Метою еволюційного прототипування** є поставка працюючої системи кінцевому користувачеві, це означає, що необхідно почати створення системи, що реалізує вимоги користувача, які найбільш зрозумілі і які мають найвищий пріоритет. Вимоги з більш низьким пріоритетом і нечіткі вимоги реалізуються по запитах користувачів.
* **Метою експериментального** **прототипування** є перевірка і формування системних вимог. Тут спочатку створюється прототип, що реалізує ті вимоги, які сформульовані нечітко і з якими необхідно “розібратися”. Вимоги, які сформульовані чітко і зрозуміло, не мають потреби в прототипуванні.

Інша важлива відмінність між цими підходами стосується керування якістю розроблювальної системи. Експериментальні прототипи мають дуже короткий строк життя. Вони швидко міняються і для них висока експлуатаційна надійність не потрібна. Для експериментального прототипу допускається знижена ефективність і безвідмовність. оскільки прототип повинен виконати тільки свою основну функцію ‒ допомогти в розумінні вимог.

На противагу цьому прототипи, які еволюціонують у закінчену систему, повинні бути розроблені з такими ж стандартами якості, що і будь-яке інше програмне забезпечення. Вони повинні мати стійку структуру і високу експлуатаційну надійність. Вони повинні бути безвідмовні, ефективні і відповідати відповідним до стандартів.

***Еволюційне прототипування***

В основі еволюційного прототипування лежить ідея розробки первісної версії системи, демонстрації її користувачам і наступної модифікації аж до одержання системи, що відповідає всім вимогам. Такий підхід спочатку використовувався для розробки систем, які важко або неможливо специфікувати (наприклад, систем штучного інтелекту). У цей час він стає основною методикою при розробці програмних систем. Еволюційне прототипування має багато спільного з методами швидкої розробки додатків і часто входить у ці методи як їх складова частина

Цей метод прототипування має дві основні переваги.

1. ***Прискорення розробки системи.*** Як вказувалося у введенні, сучасні темпи змін у діловій сфері вимагають швидких змін програмного забезпечення. У деяких випадках швидка поставка ПЗ, зручність і простота його використання більш важливі, чим повний спектр функціональних можливостей системи або довгострокові можливості її супроводу.
2. ***Взаємодія користувача із системою.*** Участь користувачів у процесі розробки означає, що в системі більш повно будуть враховані користувацькі вимоги.

***Експериментальне прототипування***

У цій моделі розширений етап аналізу вимог з метою зменшення загальних витрат на розробку. Основне призначення прототипу ‒ зробити зрозумілими вимоги і надати додаткову інформацію для оцінки ризиків. Після цього прототип більше не використовується і не бере участь в подальшому процесі розробки системи.

Цей метод прототипування зазвичай використовується для розробки апаратних систем. Перш ніж буде почата дорога розробка системи, створюється макет (прототип), який використовується для перевірки структури системи. Електронний макет системи створюється з використанням готових компонентів, що дозволяє розробити версію системи до того, як будуть вкладені грошові кошти в розробку спеціалізованих інтегральних схем.

Експериментальний прототип програмних систем зазвичай не використовується для перевірки архітектури системи, він допомагає розробити системні вимоги. Прототип часто зовсім не схожий на кінцеву систему. Система розробляється по можливості швидко, тому для прискорення формування вимог використовується спрощений прототип системи. Б експериментальний прототип закладаються тільки обов'язкові системні функції, стандарти якості для прототипу можуть бути знижені, критерії ефективності ігноруються. Мова програмування прототипу часто відрізняється від мови програмування, на якій буде створюватися остаточний варіант системи.

**Виконання роботи**

1. На основі раніше розробленого технічного завдання та вимог до програмного продукту підготувати блок-схему функціонування прототипу програмного продукту. У блок-схемі представити структурні елементи (модулі) та їх функціональне призначення.
2. Подати блок-схему на перевірку викладачу.
3. Захистити роботу, відповівши на контрольні запитання.

**Контрольні питання**

1. Що таке прототип програмного забезпечення?
2. На яких етапах розробки програмного продукту актуальне використання прототипу та як саме?
3. Для яких цілей може бути використаний прототип?
4. Наведіть переваги використання прототипу.
5. Що таке еволюційне прототипування та яка його мета?
6. Що таке експериментальне прототипування та яка його мета?
7. Які відмінності між еволюційним та експериментальним прототипуванням?

Програмний продукт

«Залізнична станція»

Поле оголошень

Виведення інформації на загальне джерело

Бігуча стрічка новин

Модуль 3. Інформаційний

Пошук по часу руху / по напрямках

Фільтрація даних

Модуль 2. Клієнтський

Сортування даних

Виведення інформації

Введення інформації про затримку потягів

Введення розкладу руху

Корегування розкладу

Авторизація персоналу

Модуль 1. Адміністративний