

СУЧАСНА РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ: ЗАВДАННЯ ТА ЇЇ ВИРІШЕННЯ

В даний час дуже велика увага приділяється розробці програмного забезпечення (ПО), яке є важко подається формалізації і одночасно є творчим процесом. Слід враховувати той фактор, що схемна реалізація випереджає програмну. З огляду на це, розробники змушені вирішувати дві основні задачі: *побудова програмного комплексу в певні терміни відповідно до вимог замовника і пошук оптимальної реалізації компонентів додатків.*

Найпопулярнішими є програми, призначені для роботи з текстами - текстові редактори і видавничі системи. Великої популярності набули програми обробки графічної інформації. Комп'ютерна графіка в даний час є однією з областей програмного забезпечення, що розвивається найбільш динамічно. Вона включає в себе введення, обробку і виведення графічної інформації - креслень, малюнків, картин, текстів тощо - засобами комп'ютерної техніки. Різні типи графічних систем дозволяють швидко будувати зображення, вводити ілюстрації за допомогою сканера або відеокамери, створювати анімаційні ролики.

Графічні редактори дозволяють користуватися різним інструментарієм художника, стандартними бібліотеками зображень, наборами стандартних шрифтів, редагуванням зображень, копіюванням і переміщенням фрагментів по сторінках екрану тощо. Для виконання розрахунків і подальшої обробки числової інформації існують спеціальні програми - електронні таблиці. У процесі діяльності будь-якого фахівця часто потрібно представити результати роботи у вигляді таблиць, де одна частина полів зайнята вихідними даними, а інша - результатами обчислень і графічного аналізу. Характерними для них є великий обсяг інформації, що переробляється, необхідність багаторазових розрахунків при зміні вихідних даних. Автоматизацією подібної рутинної роботи і займаються електронні таблиці.

У зв'язку з тим що галузь інженерії ПО є однією з найбільш динамічно щорічно розвиваючої в сфері ІТ (від англ. *Information Technology*) інформаційних технологій, *що включає в себе процеси, методи пошуку, збору, зберігання, обробки, надання, поширення інформації та способи здійснення таких процесів і методів активно розвиваються з 1960-х років* і демонструє суттєвий щорічний приріст обсягів ринку реалізації власної продукції, важливо розглянути технології, здатні перетворювати «процес творчого пошуку» в «інженерну науку» з усіма властивими їй атрибутами.

При розробці ПО орієнтуємося на основні складові: продукт, персонал, процес, проект, артефакти. Проаналізуємо наведені їх:

Продукт – то, що має бути отримано в результаті діяльності розробників ПЗ. До продукту спочатку пред'являються певні вимоги, які лімітують «політ фантазії» розробників. Продукт складається з програмного коду, документації і різних артефактів.

Персонал – в першу чергу, це колектив розробників різної спеціалізації. Крім того, до персоналу відносять замовників, користувачів, інвесторів.

Процес – послідовність етапів аналізу, проектування, розробки, впровадження та підтримки в ході створення продукту.

Проект – сукупність дій, необхідна для створення продукту.

Артефакти – додаткові результати, отримані в ході розробки ПО, очікувані або отримані випадковим чином.

Звернемо увагу на істотні аспекти проектування.

Найбільш поширеною парадигмою аналізу, проектування і програмування на поточний момент є об'єктно-орієнтована парадигма. Більшість сучасних RAD-середовищ (Rapid Application Development) побудовані з використанням саме такої парадигми.

При розробці ПЗ (так само як і при реалізації інших проектів) потрібне застосування стандартів, що дозволяють реалізувати не просто один окремо взятий проект, а створити конвеєр по розробці. Як правило, під подібними стандартами розуміють технології, прийняті індустрією і ринком - промислові стандарти. Як стандарт на інструментарій опису моделей при розробці ПЗ на поточний момент широко використовується уніфікована мова моделювання UML (*Unified Modeling Language*).

Особливо слід відзначити два факти:

1. UML не методологія, а нотація опису моделей;

2. UML є розширеною концепцією опису моделей не тільки в області програмування, але і в області системного аналізу, моделювання технічних і економічних систем, а також їх динаміки (стандарт-діалект SysML (*System Modeling Language*)).

У даній статті за основу опису методів використання UML при розробці ПЗ обраний колективний процес розробки програмного забезпечення TSP (Team Software Process). Необхідність у використанні TSP-методології замість індивідуального процесу розробки PSP (Personal Software Process) обумовлена неможливістю роботи одного програміста при розробці сучасного ПО. Розглянемо основні проблеми, які повинні бути вирішені за допомогою TSP:

1. Збір команди розробників - група повинна бути самокерованою, для цього необхідно визначити власні цілі, скласти процес і плани досягнення поставлених цілей, а також відстежувати хід виконання окремих етапів робіт;

2. Дати відповідь керівнику проекту про спосіб управління командами: забезпечити узгодження діяльності команд, надати учасникам персональні інструкції, мотивувати до якнайшвидшого досягнення результатів і забезпечити максимальну продуктивність до завершення проекту;

3. Забезпечити впевнене досягнення рівнів моделі зрілості можливостей CMM (Capability Maturity Model);

4. Визначити напрямки і способи поліпшення процесу розробки для успішних команд;

5. Сприяти підготовці кваліфікованих фахівців для роботи в ІТ-галузі. В якості основних описів процесу розробки використовується модель уніфікованого процесу розробки програмного забезпечення USDP (Unified Software Development Process), що включає п'ять стадій: *попередній аналіз вимог, детальний аналіз вимог, проектування, реалізація, тестування*. З точки зору фактора часу, процес проходить чотири фази: початок, проектування, конструювання, перехід.

Відповідно до USDP виділяються наступні моделі: *використання, аналітична, проектування, розгортання, реалізації та тестування*.

З точки зору фактора часу, процес проходить наступні фази: *початок, проектування, конструювання, перехід*. Фази проектування, конструювання та переходу - багато ітеративні. Відповідно до USDP виділяються наступні моделі: *використання, аналітична, проектування, розгортання, реалізації та тестування*.

У статті пропонується дев'ять послідовностей етапів розробки програмного проекту: визначення концепції системи; аналіз предметної області; проектування системи: проектування класів; реалізація; тестування; навчання персоналу; розгортання і підтримка.

Процес розробки-безперервний. Моделі постійно уточнюються і оптимізуються, навіть у випадках переходу від аналізу до проектування, а потім до реалізації. На всіх етапах розробки ПЗ використовують одні і ті ж вибрані спочатку концепції проектування і системи позначень. Різниця в тому, що спочатку процесу увагу в основному приділяється потребам замовника, а в кінці - обчислювальних ресурсів.

Об'єктно-орієнтований підхід до побудови ПО переносить основні зусилля з розробки програмного забезпечення на етапи аналізу і проектування. У ряді ситуацій може виникнути думка про недоцільність витрат істотного обсягу часу на аналіз і проектування системи. Однак ці витрати окупаються швидкою і простою реалізацією, так як побудоване відповідно до об'єктно-орієнтованої концепції програмування програмне забезпечення виявляється більш зрозумілим і легко адаптованим, простіше здійснювати його підтримку і вносити зміни відповідно до змін реального світу.

Висновки. У даній статті розглядаються питання аналізу, проектування і розробки. Питання тестування, навчання, розгортання і підтримки детально не розглядаються, так як вони виходять за рамки безпосередньо проектних робіт, а відносяться, скоріше, до питань проектування інформаційних систем. Питання тестування складають набір спеціалізованих технологій і методик, що також не дозволяє їх розглянути в рамках одного матеріалу.