



Архітектура та проектування програмного забезпечення



Архітектура та проектування програмного забезпечення

Зміст

1. Визначення архітектури ПЗ
2. Важливість архітектури
3. Проектування систем
4. Мета архітектури
5. Фактори, що впливають на формування архітектури
6. Принципи проектування архітектури
7. Тестування архітектури



Створення архітектури застосування

Створення архітектури застосування – це процес формування структурного рішення, яке відповідає всім технічним і операційним вимогам і забезпечує оптимальні загальні атрибути якості, такі як продуктивність, безпека та керованість.

Воно включає прийняття ряду рішень, кожне з яких може мати суттєвий вплив на якість, продуктивність, безпеку, зручність використання і загальний успіх застосування.



Визначення архітектури ПЗ

Відповідно до визначення архітектури ПЗ фахівцями: Філіп Крачтен, Курт Бітнер, Річ Рейтман в роботі Мері Шоу і Девіда Гарлана: Архітектура ПЗ включає в себе ряд важливих рішень про організацію програмної системи, серед яких вибір структурних елементів і їх інтерфейсів, які складають і об'єднують систему в єдине ціле; поведінку яка забезпечується спільною роботою цих елементів; організацію цих структурних і поведінокових елементів в більш крупні підсистеми, а також архітектурний стиль, якого притримується дана організація. Вибір архітектури також стосується функціональності, зручності використання, стійкості, продуктивності, повторного використання, зрозумілості, економічних та технологічних обмежень, естетичного сприйняття і пошуку компромісів.



Визначення архітектури ПЗ

У книзі “Архітектура корпоративних програмних застосувань” Мартіна Фаулера, розглядаючи проектування архітектури, виділяється декілька загальних елементів:

- розділення системи на складові частини в самому першому приближенні, прийняття рішень, які важко змінити згодом;
- відпрацювання множини можливих варіантів архітектури для системи;
- важливість з точки зору архітектури різних аспектів може змінюватися в процесі життєвого циклу системи;
- і в кінці кінців під архітектурою можна розуміти те, що має значення.



Визначення архітектури ПЗ

У книзі “Архітектура ПЗ на практиці” Басс, Клементс і Казман дають таке визначення:

Архітектура програмної чи обчислювальної системи – це структура чи структури системи, які включають програмні елементи, видимі ззовні властивості цих елементів і взаємовідношення між ними.

Архітектура стосується зовнішньої частини інтерфейсів, внутрішні деталі елементів – деталі, які відносяться виключно до внутрішньої реалізації – не являються архітектурними.



Важливість архітектури

ПЗ повинно будуватися на міцному фундаменті.

Неправильне визначення ключових сценаріїв, неправильне проектування загальних питань чи нездатність виявляти довгострокові наслідки основних рішень можуть поставити під загрозу все застосування.

Неправильно відпрацьована архітектура обумовлює нестабільність ПЗ, неможливість підтримання існуючих чи майбутніх бізнес вимог, складність при розгортанні чи керуванні в середовищі виробничої експлуатації.



Проектування систем

Проектування систем повинно здійснюватися з урахуванням вимог користувача, системи (ІТ-інфраструктури) і бізнес цілей. Необхідно шукати компроміси і знаходити баланс між конкуруючими вимогами цих трьох областей.





Проектування систем

Вибір структур даних і алгоритмів їх обробки або деталей реалізації окремих компонентів є питаннями проектування.

Часто питання архітектури і проектування перетинаються. Замість того щоб вводити жорсткі правила, що розмежовують архітектуру і проектування, має сенс комбінувати ці дві області.

Наприклад: закупку нового обладнання, здатного оброблювати інформацію швидше можна замінити новим алгоритмом обробки інформації, який здійснюватиме обробку швидше.



Проектування систем

Питання при розробці архітектури системи:

Як користувач буде використовувати застосування?

Як застосування буде розгортатися і обслуговуватися при експлуатації?

Які вимоги по атрибутам якості, таким як безпека, продуктивність, можливість паралельної обробки, інтернаціоналізація та конфігурація, будуть висуватися до застосування?

Як спроектувати застосування, щоб воно залишалось гнучким і зручним в обслуговуванні протягом довгого часу?

Основні архітектурні напрямки, які можуть впливати на застосування зараз або після його розгортання?



Мета архітектури

Архітектура програми повинна об'єднувати бізнес-вимоги і технічні вимоги через розуміння варіантів використання з подальшим перебуванням шляхів їх реалізації.

Мета архітектури - виявити вимоги, що впливають на структуру програми.

Гарна архітектура знижує бізнес-ризики, пов'язані зі створенням технічного рішення.

Хороша структура володіє значною гнучкістю, щоб справлятися з природним розвитком технологій, як в області устаткування і ПЗ, так і призначених для користувача сценаріїв і вимог.

Архітектор повинен враховувати загальний ефект від прийнятих проектних рішень, обов'язково присутні компроміси між атрибутами якості (такими як продуктивність і безпека) і компроміси, необхідні для виконання призначених для користувача, системних і бізнес-вимог.



Мета архітектури

Архітектура повинна:

Розкривати структуру системи, але приховувати деталі реалізації.

Реалізовувати всі варіанти використання і сценарії.

По можливості відповідати всім вимогам різних зацікавлених сторін.

Виконувати вимоги, як по функціональності, так і за якістю.



Фактори, що впливають на формування архітектури

Фактори, що впливають на формування архітектури визначаються побажаннями користувача, вимогами бізнесу (швидкі результати, зміна стилю робочих процесів), а також адаптованістю дизайну.

Фактори:

Наділення користувача повноваженнями.

Дизайн, що підтримує наділення користувача повноваженнями (їх зміну, видалення) є гнучким, здатним до налаштування і орієнтованим на користувача.

Не диктуйте, а дозволяйте користувачу визначати користувачу стиль взаємодії із застосуванням, але при цьому не перевантажуйте їх непотрібними опціями і налаштуваннями, що може збити з пантелику.



Фактори, що впливають на формування архітектури

Зрілість ринку.

Використовуйте переваги зрілості ринку, застосовуючи існуючі платформи і технології.

Створюйте додатки в настільки високого рівня середовищах, наскільки це дозволяють вимоги, що пред'являються до застосувань, це дозволить сконцентруватися на тому, що дійсно унікально для вашого застосування, а не на відтворенні вже існуючих функцій.

Використовуйте шаблони, які є багатими джерелами перевірених рішень поширених проблем.



Фактори, що впливають на формування архітектури

Гнучкий дизайн.

Все більшої популярності набувають гнучкі дизайни, що використовують слабе зв'язування (взаємозалежність), що робить можливим повторне використання і спрощує підтримку.

Архітектура з можливістю підключення модулів дозволяє реалізувати розширюваність після розгортання.

Також для забезпечення взаємодії з іншими системами можна використовувати переваги таких сервіс-орієнтованих технік, як SOA.



Фактори, що впливають на формування архітектури

Тенденції.

При побудові архітектури необхідно розуміти тенденції, які можуть вплинути на дизайн після розгортання.

Наприклад тенденції щодо:

використання насичених UI і мультимедіа;

використання застосувань, що складаються з інших застосувань,

збільшення пропускної здатності та доступності мереж,

все більшого поширення мобільних пристроїв,

тривалого зростання продуктивності устаткування,

інтересу до моделей блогів і спільнот,

зростання популярності обчислень в хмарі і віддаленої роботи.



Принципи проектування архітектури

На даний час при продумуванні архітектури ми припускаємо, що наш дизайн буде еволюціонувати згодом і що абсолютно неможливо наперед знати все те, що необхідно для проектування системи.

При проектуванні архітектури необхідно відповісти на такі питання:
Які частини архітектури є фундаментальними, зміни яких, у випадку невірної реалізації представляють найбільші ризики?
Які частини архітектури наймовірніше зазнають змін?, а також
Проектування яких частин можна відкласти?
Основні припущення, і як вони будуть перевірятися?
Які умови можуть привести до реструктуризації дизайну?

Не намагайтесь створювати дуже складну архітектуру і не робіть припущень, які ви не зможете переврити.

Деякі аспекти дизайну повинні бути приведені до порядку на ранніх стадіях процесу, тому, що їх можливе переопрацювання може потребувати суттєвих витрат. Такі області треба виявляти якомога раніше і приділяти їм достатньо уваги.



Принципи проектування архітектури

Принципи проектування архітектури:

Створюйте, щоб змінювати.

Продумайте, як згодом може знадобитися змінити додаток, щоб воно відповідало знову виникаючим вимогам і задачам, і передбачте необхідну гнучкість.

Створюйте моделі для аналізу і скорочення ризиків.

Використовуйте засоби проектування, системи моделювання, такі як Уніфікована мова моделювання (Unified Modeling Language, UML), і засоби візуалізації, коли необхідно виявити вимоги, прийняти архітектурні та проектні рішення і проаналізувати їх наслідки. Тим не менш, не створюйте занадто формалізовану модель, вона може обмежити можливості для виконання ітерацій і адаптації дизайну.



Принципи проектування архітектури

Використовуйте моделі і візуалізації як засіб спілкування під час спільної роботи.

Для побудови гарної архітектури критично важливо мати ефективний обмін інформацією про дизайн, прийняті рішення та внесені зміни.

Використовуйте моделі, представлення (подання) та інші способи візуалізації архітектури для ефективного обміну інформацією та зв'язки з усіма зацікавленими сторонами, а також для забезпечення швидкого оповіщення про зміни в дизайні.

Виявляйте ключові інженерні рішення.

Намагайтесь визначити ключові інженерні рішення і області, в яких найчастіше виникають помилки. На самому початку проекту приділіть достатньо часу і уваги для прийняття правильних рішень, це забезпечить створення більш гнучкого дизайну, внесення змін до якого не буде потребувати повної його переробки.



Тестування архітектури

При тестуванні архітектури дайте відповіді на такі питання:

Які допущення було зроблено в цій архітектурі?

Яким явним або неявним вимогам, що маються на увазі відповідає дана архітектура?

Основні ризики при використанні такого архітектурного рішення?

Які заходи протидії для зниження основних ризиків?

Чи є дана архітектура поліпшенням базової архітектури або одним з можливих варіантів архітектури?



Архітектура та проектування програмного забезпечення

Дякую за увагу