

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ „ЛЬВІВСЬКА ПОЛІТЕХНІКА”

**СПЕЦИФІКАЦІЯ ВИМОГ
ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт № 1, 2 з дисципліни
„Аналіз вимог до програмного забезпечення”
студентів базового напрямку
6.050103 „Програмна інженерія”

*Затверджено
на засіданні кафедри
програмного забезпечення
Протокол № 1 від 28.08.2015 р.*

Львів – 2015

Специфікація вимог до програмного забезпечення: Методичні вказівки до лабораторних робіт № 1, 2 з дисципліни „Аналіз вимог до програмного забезпечення” для студентів базового напрямку 6.050103 „Програмна інженерія” / Укл.: О.Є. Білас, Н. Б. Мельник – Львів: Видавництво Національного університету „Львівська політехніка”, 2015. – 16 с.

Укладачі

Білас О.Є., доцент кафедри ПЗ
Мельник Н.Б., ст.викладач кафедри ПЗ

Відповідальний за випуск

Федасюк Д.В., д-р тех. наук, проф.

Рецензенти

Вступ

Аналіз вимог - процес вивчення потреб і цілей користувачів, класифікація і перетворення їх на вимоги до системи, апаратури і ПЗ, встановлення і вирішення конфліктів між вимогами, визначення пріоритетів, меж системи і принципів взаємодії із середовищем функціонування.

Специфікація вимог до програмного забезпечення (англ. *Software Requirements Specification (SRS)*) - процес формалізованого опису функціональних і нефункціональних вимог, вимог до характеристик, які будуть імплементуватися, втілюватися на етапах ЖЦ ПЗ. Це повний опис поведінки системи, що розробляється.

У специфікації вимог відображається структура ПЗ, вимоги до функцій, якості і документації. Специфікуються також системні вимоги, нефункціональні (додаткові) вимоги і вимоги до взаємодії з іншими компонентами і платформами (БД, СКБД, мережа й ін.)

Специфікація вимог

Вимоги до продукту охоплюють очікування користувачів щодо зовнішньої поведінки системи і погляди розробників на деякі параметри системи.

Вимоги до ПЗ містять у собі системні, функціональні і нефункціональні вимоги.

Вимоги користувачів (*user requirements*) віддзеркалюють вимоги споживачів до спектра розв'язуваних майбутньою системою задач. Вони подаються як текстовий опис або сценарії, прецеденти, таблиці «подія-відгук» тощо.

Системні вимоги (*system requirements*) визначають зовнішні умови виконання системних функцій і обмежень на створення продукту, а також вимоги до опису програмно-апаратних підсистем. Системні вимоги накладають обмеження на архітектуру системи, засоби її візуального подання і функціонування. Для опису системних вимог використовують спеціальні шаблони і форми, що допомагають уявити вхідні і вихідні дані й автоматизувати ці вимоги.

Вимоги до атрибутів якості (*quality attributes*) - це деякі обмеження на властивості функцій або системи, важливі для користувачів або розробників. Наприклад, зручність, цілісність, стійкість системи до збоїв.

Функціональні вимоги - це перелік функцій або сервісів, які повинна надавати система, а також обмежень на дані і поведінку системи при їхньому

виконанні. Специфікація функціональних вимог - опис функцій та їхніх властивостей, які не містять у собі протиріч і виключень.

Нефункціональні вимоги визначають умови виконання функцій (наприклад, захист інформації у БД, аутентифікація доступу до ПС тощо), середовища, що безпосередньо не пов'язані з функціями, а відбивають потреби користувачів щодо їх виконання. Ці вимоги характеризують принципи взаємодії із середовищами або іншими системами, а також визначають показники часу роботи, захисту даних і досягнення якості з урахуванням рекомендацій використовуваного стандарту. Вони можуть встановлюватися як числові значення (наприклад, час чекання відповіді, кількість клієнтів, що обслуговуються і ін.) у різних одиницях виміру, включаючи, наприклад, ймовірність (значення ймовірності безвідмовної роботи системи – показника її надійності).

Основні вимоги до ПЗ

Для більшості сучасних програмних систем вимоги складаються з наступних умов та обмежень:

- склад виконуваних функцій системи (запуск, швидкість реакції й ін.);
- конфіденційність, безпека і захист даних;
- відмовостійкість;
- одночасність доступу користувачів до системи;
- час очікування відповіді при звертанні до системи (продуктивність);
- положення стандартів з виконання сформульованих вимог.

Дані вимоги визначаються і формалізуються аналітиками на етапі аналізу і проектування структури системи. Так, у випадку вимог з безпеки функціонування системи, в системі виділяються категорії користувачів, що мають право доступу до тих або інших функцій (програмних компонентів) або даних, та передбачаються додаткові функції системи з перевірки доступу (санкціонований доступ до них чи ні). Якщо потрібно обмежити доступ до конкретних даних (наприклад, до окремих записів, полів у таблиці), то в системі може передбачатися, наприклад, мандатний захист. Для захисту всієї системи від несанкціонованого доступу користувачі реєструються і проходять аутентифікацію для роботи із системою.

Для відновлення і збереження резервних копій БД, архівів баз даних аналізуються можливості СУБД і способи забезпечення необхідного рівня безперебійної роботи системи, правил доступу авторизованих користувачів і

заходів боротьби з різними загрозами, що надходять ззовні від користувачів, які не мають прав доступу до деяких або до всіх даних системи.

Для вихідного продукту пред'являються нефункціональні вимоги до:

- практичності (якість інтерфейсу, швидкості вивчення продукту й ін.);
- продуктивності (пропускна здатність, час реакції й ін.);
- надійності виконання (без помилок і відмов);
- зовнішніх інтерфейсів, за якими виконується взаємодія з іншими компонентами або підсистемами.

Опис усіх видів вимог проводиться з урахуванням стандартів, наприклад, стандарту з якості ISO/IEC ДСТУ 9126 і стандартизованого термінологічного довідника, що містить у собі загальноприйняті терміни щодо структури ПЗ і призначення функцій системи. Специфікація вимог відображає принципи взаємодії проектованої системи з іншими середовищами, платформами і загальносистемним забезпеченням (БД, СКБД, мережі та ін.).

Формування документа зі специфікаціями вимог завершується на етапі проектування архітектури, після чого він узгоджується з замовником системи і використовується як керування дій при виконанні задач розробки програмного продукту на етапах ЖЦ і отриманні готового продукту. При виявленні на цих етапах неузгоджених вимог, проводиться їхнє уточнення і, відповідно, вносяться зміни у деякі задачі процесу розроблення системи або характеристики продукту.

Виявлення вимог

У сучасних технологіях процес ЖЦ, у якому фіксуються вимоги до розробки системи, є визначальним для задання функцій, термінів і вартості робіт, а також показників якості, які необхідно досягти в процесі розроблення. Виявлення вимог проводиться під час обговорення проекту, аналізу особливостей предметної області і визначення підходів до її проектування на етапах ЖЦ.

Вимоги відбивають потреби людей (замовників, користувачів, розробників), зацікавлених у створенні ПС. Замовник і розробник спільно обговорюють проблеми проекту, збирають вимоги, проводять їхній аналіз, перегляд і визначають необхідні обмеження.

Обговорення проекту системи відбувається з метою вивчення думки і вироблення перших висновків щодо доцільності виконання проекту і прогнозування реальності його виконання в заданий термін і за кошти, що дає замовник. Природно, особа, яка замовила проект системи, бажає отримати від

розробника набір необхідних послуг, за якими будуть звертатися різні категорії користувачів: оператори, менеджери, фахівці з ПЗ.

Розробники системи можуть оцінити можливість реалізації проекту системи, що замовляється, у заданий термін і бюджет. Серед розробників призначаються головний аналітик, відповідальний за вимоги до системи, і провідний програміст, відповідальний за їхню реалізацію. Вони узгоджують вимоги і визначають сферу дії проекту на спільних переговорах із замовником.

В обговоренні вимог до системи беруть участь:

- представники замовника з декількох професійних груп;
- кінцеві користувачі програмного продукту;
- аналітики і розробники майбутньої системи.

Погоджена сфера дій у проекті дає можливість оцінити необхідні інвестиції в проект, заздалегідь визначити можливі ризики і здатності розробників щодо виконання проекту. Підсумком обговорення проекту може бути рішення про розгортання реалізаційних робіт по проекту або відмови від нього.

Розробники вимог повинні мати відповідні знання в даній предметній області і вмінні аналізувати проблему, потреби замовника і користувачів системи, виявляти функції системи, що мають бути реалізовані в проекті, вносити зміни в окремі елементи вимог у процесі їх виконання.

У вимогах до ПС, крім проблем системи, формулюються реальні потреби замовника щодо функціональних, операційних і сервісних можливостей майбутньої системи. Результати дії дослідження й аналізу предметної області фіксуються в документі з опису вимог і в договорі між замовником і виконавцем проекту.

Помилки через нечіткі або неоднозначні формулювання вимог можуть призвести до того, що виготовлена система не буде задовольняти замовника. Тому на етапах розробки вимоги повинні постійно уточнюватися і знову затверджуватися замовником. В окремих випадках внесені зміни у вимоги можуть обумовити необхідність перепроєктування окремих частини або всієї системи в цілому. Відповідно до статистики, частка помилок у постановці вимог і у визначенні задач системи перевищує частку помилок, що допускається під час кодування системи. Це обумовлюється суб'єктивним характером процесу формулювання вимог і відсутністю способів їхньої формалізації.

Добування вимог

Джерела відомостей для добування вимог:

- мета і задачі проекту, що формулюються замовником майбутньої системи, і які обговорюються розробниками;
- колектив, який реалізує функціонал системи.

Вивчення і фіксація реалізованих функціональних можливостей у діючій системі є підґрунтям для накопичення досвіду для формулювання нових вимог до неї. При цьому необхідно відокремлювати нові вимоги до системи від старих вимог, щоб не повторити невдалі розв'язки щодо старої системи в новому її виконанні.

Вимоги до системи формулюються замовником у термінах понять його предметної області з урахуванням відомих словників, стандартів, існуючих умов середовища функціонування майбутньої системи, а також трудових і фінансових ресурсів, виділених на розробку системи.

Методи збирання вимог:

- інтерв'ю з виразниками інтересів замовника системи;
- вивчення прикладів можливих варіантів виконання функцій, ролей відповідальних осіб, які пропонують ці варіанти, або тих, що взаємодіють із системою при її функціонуванні;
- спостереження за роботою діючої системи для відокремлення властивостей, що обумовлені кадровими аспектами.

Зовнішні і внутрішні аспекти вимог пов'язують з характеристиками якості і відносяться до властивостей створюваного продукту, а саме, функцій системи, її призначення і виконання в заданому середовищі. Наприкінці користувач очікує досягнення максимального ефекту від застосування вихідного продукту та орієнтується на його кінцеву експлуатаційну якість.

Отримання зовнішніх і внутрішніх характеристик якості досягається спеціально розробленими методами з виконання процесів ЖЦ. Остаточні сформульовані вимоги - основа для підпису контракту між замовником і розробником системи.

Основні джерела отримання інформації про потреби клієнтів:

- опитування потенційних користувачів;
- документи, де описаний уже працюючий або конкуруючий продукт;
- специфікація вимог до системи вищого рівня;
- звіти про помилки від служби підтримки;
- маркетингові дослідження і опитування;

- спостереження за користувачами на робочих місцях;
- сценарії аналізу задач користувача;
- події і реакції на них.

Для отримання якісного програмного продукту, що відповідатиме запитам замовників, дуже важливе значення має етап збирання вимог. До цієї роботи повинні бути залучені всі зацікавлені в проекті особи, а саме:

- замовники, які фінансують проект;
- користувачі, які безпосередньо працюють з додатком (підклас замовників);
- аналітики вимог, які пишуть вимоги і передають їх розробникам ;
- розробники, які створюють, розгортають і підтримують продукт;
- тестувальники, які визначають відповідність поведінки ПЗ бажаному;
- технічні письменники, які відповідають за створення керівництва користувачів, тренувальних матеріалів та довідкової системи;
- менеджер по проекту, який планує процес і керує командою розробників аж до успішного випуску продукту ;
- співробітники правового відділу, які стежать, щоб продукт не суперечив чинним законам і постановам;
- співробітники відділу продажів і маркетингу, виїзної служби підтримки та всі ті, кому доведеться працювати з продуктом і його користувачами.

Характеристики якісних вимог

- Повнота. Вимога повинна повністю описувати функціональність.
- Коректність. Вимога повинна точно описувати функціональність.
- Реалізовуваність. Вимога повинна бути реалізовуваною.
- Необхідність. Вимога повинна відображати те, що потрібно користувачу.
- Пріоритетність. Вимоги повинні бути розставленими за пріоритетами.
- Недвозначність. Вимога повинна однозначно інтерпретуватись.
- Тестованість. Вимога повинна легко перевірятись.

Лабораторна робота 1

Тема: Аналіз існуючих програмних систем (програм-аналогів).
Відновлення вимог з програм-аналогів для обраної предметної області^{*1}.

Мета: Проаналізувати існуючі програмні продукти заданої предметної області.

Завдання. Здійснити порівняльний аналіз програм-аналогів, провести відновлення вимог з програм-аналогів.

Порядок виконання лабораторної роботи

1. Вибрати предметну область із запропонованого переліку.
2. Здійснити пошук в мережі Інтернет 5-ти існуючих програм-аналогів (сайтів) для обраної предметної області (вітчизняні та закордонні ресурси).
3. Описати кожну із розглянутих програмних систем за наступним планом:
 - назва, компанія-виробник;
 - основні функціональні можливості;
 - переваги ПС;
 - недоліки.
4. Здійснити порівняння існуючих програм-аналогів і внести дані в таблицю:

	ПС 1	ПС 2	...	
Властивість 1	+	+		
Властивість 2	-	+		
...				

5. На основі проведеного аналізу скласти перелік вимог до ПС для заданої предметної області.

6. Оформити звіт.

¹ Предметна область (тема) призначається студенту індивідуально, відповідно до доданого списку тем.

Лабораторна робота 2

Тема: Специфікація вимог до програмного продукту для обраної предметної області*.

Мета: Розробити специфікацію вимог та оформити її згідно поданого нижче зразка.

Специфікація вимог до програмного продукту (Software Requirements Specification, SRS) для <Назва проекту> (Зразок)

Вступ

Призначення, мета

<Визначити продукт, вимоги до якого описані в цьому документі. Описати межі продукту, зокрема якщо цей документ описує лише частину системи чи окрему підсистему.>

Посилання

<Список документів чи Web адрес, на які посилається цей документ. Може включати зразки користувацьких інтерфейсів, контракти, стандарти, варіанти використання.>

Загальний опис

Перспективи продукту

<Описати контекст і витоки продукту, який специфікується цим документом. Наприклад, стан продукту, як члена сімейства продуктів, заміна існуючих систем, чи новий самодостатній продукт. Якщо SRS визначає компонент великої системи, пов'язати вимоги великої системи із функціональністю цього продукту і визначити інтерфейси між ними. Ефективними є прості діаграми, що показують основні компоненти загальної системи, взаємозв'язок підсистем чи зовнішні.>

Характеристики продукту

<Резюмувати основні характеристики продукту чи істотні функції, що він здійснює чи дозволяє здійснювати користувачу. Деталі представляються в Розділі 3, тому тут потрібне узагальнення вищого рівня. Організувати функції, щоб вони були зрозумілими будь-якому читачу документа. Ефективним є

представлення основних груп пов'язаних вимог і їхніх зв'язків діаграмами потоків даних чи діаграмами класів.>

Класи користувачів та їх характеристики

<Визначити різні класи користувачів, які як ви очікуєте, будуть використовувати продукт. Класи користувачів можуть диференціюватись, базуючись на частоті використання, підмножині функцій продукту, яка використовується, технічній експертизі, рівнях безпеки чи привілеїв, рівню освіти чи досвіду. Описати доцільні характеристики кожного класу користувачів. Відділити пріоритетні класи користувачів від тих, що є менш важливими.>

Середовище функціонування

<Описати середовище, в якому буде функціонувати продукт, включаючи апаратну платформу, операційну систему і версії, і будь-які інші програмні компоненти чи аплікації, з якими воно має коректно співіснувати.>

Обмеження проектування і реалізації

<Описати питання чи пункти, що будуть обмежувати можливості, доступні розробникам. Може містити: корпоративні чи регуляторні правила; апаратні обмеження (часові вимоги, вимоги щодо пам'яті); взаємодії з іншими аплікаціями; специфічні технології, інструменти, бази даних; паралельні операції; мовні вимоги; протоколи комунікацій; обговорення безпеки; проектні програмні стандарти (наприклад, якщо організація замовника буде відповідальною за супровід продукту).>

Документація користувача

<Список компонент документації користувача (такі як керівництво по використанню, on-line допомога, чи інструкції), що буде надаватися разом з програмним продуктом. Визначити формати та стандарти документації.>

Припущення та залежності

<Список припущень (які суперечать відомим фактам), що можуть мати вплив на вимоги встановлені в SRS. Може включати комерційні компоненти, які планується використовувати, факти щодо розробки середовища функціонування. Визначити залежність проекту від зовнішніх факторів, таких

як програмні компоненти з інших проектів, які планується повторно використовувати.>

Характеристики системи

<Ця частина ілюструє організацію функціональних вимог до продукту через характеристики системи, основні сервіси, які надає продукт. Це розділ можна організувати через варіанти використання, режими операцій, користувачькі класи, об'єкти класів, функціональну ієрархію, чи їх комбінації, залежно від того, що найбільш логічно придатно для продукту.>

Характеристика системи 1 (2,3)

<Не записуйте “Характеристика системи 1”. Задавайте конкретну, змістовну назву характеристики кількома словами.>

3.1.1 Опис і пріоритет

<Надайте короткий опис характеристики і відзначте, якого вона пріоритету Високого, Середнього, чи Низького. Можна також включати специфічні оцінки компоненти, такі як користь, штраф, ціна чи ризик (кожна оцінена у відповідній шкалі від 1 до 9).>

3.1.2 Послідовності дія/відгук

<Список послідовностей дій користувача і відгуків системи, що спричиняють режим визначений для цієї характеристики. Це відповідає елементам діалогу асоційованим з варіантами використання.>

2.1.3 Функціональні вимоги

<Перелічити детальні функціональні вимоги асоційовані із цією характеристикою. Це можливості продукту, які мають бути реалізовані, щоб користувач скористався сервісами чи виконав варіант використання. Включаючи, як продукт повинен реагувати на помилкові умови чи неправильні введення. Вимоги мають бути короткими, повними, недвозначними, верифіковуваними і необхідними.>

<Кожна вимога має бути унікально ідентифікована номером чи значущою міткою певного виду, наприклад

REQ-1:

REQ-2: .>

Вимоги зовнішніх інтерфейсів

Користувацькі інтерфейси

<Описати логічні характеристики кожного інтерфейсу між ПЗ та користувачами. Може включати зразки зображень екрану, GUI стандарти чи керівництва стилів сімейства продуктів, яких треба дотримуватись, розмітка екрану, стандарти кнопок і функцій (наприклад, допомога), що з'являються на кожному вікні, комбінації клавіш, стандарти відображення повідомлень про помилки і т.п. Визначити програмні компоненти, для яких потрібні користувацькі інтерфейси. Деталі розробки користувацького інтерфейсу мають бути документовані в окремій специфікації користувацького інтерфейсу.>

Апаратні інтерфейси

<Описати логічні та фізичні характеристики кожного інтерфейсу між ПЗ та апаратними компонентами системи. Може включати типи підтримуваних пристроїв, природу даних і керуючих взаємодій між ПЗ та апаратними засобами і комунікаційні протоколи, які будуть використані.>

Програмні інтерфейси

<Описати зв'язок між продуктом і іншими специфічними програмними компонентами (назва і версія), включаючи бази даних, операційні системи, інструменти, бібліотеки і інтегровані комерційні компоненти. Визначити дані і повідомлення, які поступають в систему і виходять з неї, описати мету кожної. Описати необхідні сервіси і природу комунікацій. Зіслатися на документи, що описують протоколи програмних інтерфейсів додатку. Визначити дані, які будуть спільно використовуватись програмними компонентами.>

Комунікаційні інтерфейси

<Описати вимоги, що пов'язані з комунікаційними функціями: електронна пошта, веб броузер, мережеві протоколи, електронні форми і т.п. Визначити прийнятні формати повідомлень. Визначити комунікаційні протоколи, які будуть використовуватись (FTP, HTTP). Визначити безпеку комунікацій чи питання шифрування, швидкість передачі даних, і механізми синхронізації.>

Інші нефункційні вимоги

Вимоги продуктивності

<Якщо є вимоги продуктивності до продукту в різних середовищах, описати їх та пояснити. Визначити часові залежності для систем реального часу. Описати такі вимоги настільки точно, як це можливо.>

Вимоги надійності

<Визначити вимоги, що пов'язані з можливими втратами, пошкодженнями, що можуть виникати при використанні продукту. Визначити заходи безпеки чи дії, які треба прийняти для запобігання цьому.>

Вимоги безпеки

<Визначити вимоги, що стосуються безпеки чи питань секретності, щодо використання продукту чи захисту даних, які використовуються чи створюються. Визначити вимоги аутентифікації користувачів.>

Атрибути якості програмного продукту

<Визначити додаткові якісні характеристики до продукту, важливі для замовників чи розробників. Наприклад, адаптовуваність, придатність, коректність, гнучкість, функціональна сумісність, супроводжуваність, портативність, надійність, стійкість, тестопридатність, зручність використання.>

Інші вимоги

<Визначити інші вимоги, що не розкриті в SRS. Це може включати вимоги бази даних, вимоги інтернаціоналізації, юридичні вимоги і т.п.>

Додаток А: Словник

<Визначити всі терміни необхідні для правильної інтерпретації SRS, включаючи аббревіатури та скорочення.>

Додаток В: Моделі аналізу

<Опційно, включити будь-які необхідні моделі аналізу, такі як діаграми потоків даних, діаграми класів, діаграми станів чи діаграми сутність-зв'язок.>

Додаток С: Список пропозицій

< Динамічний список відкритих пунктів вимог, які потрібно вирішити, незакінчені рішення, інформація, яка є потрібною, конфлікти, які очікують розв'язання.>

Контрольні питання для самоперевірки

1. Що таке вимоги до програмного забезпечення?
2. Дайте означення специфікації вимог до ПЗ.
3. Які ви знаєте основні типи вимог до ПЗ?
4. Назвіть характеристики якісних вимог.
5. В чому полягають вимоги користувачів?
6. Що таке функціональні вимоги?
7. В чому полягають нефункціональні вимоги?
8. Назвіть основні джерела отримання інформації про потреби клієнтів.
9. Розкажіть про методи збирання вимог.

Список використаної та рекомендованої літератури

1. Андон П.І., Коваль Г.І., Коротун Т.М., Лаврищева Е.М., Суслов В.Ю.
2. Основы качества программных систем.–К.: Академперіодика, 2007.– 860с.
3. Бабенко Л.П., Лаврищева К.М. Основы програмної інженерії.–К.: Знання, 2001.–269с.
4. Богданов В.В. Управление проектами в Microsoft Project 2002. Автоматизированный менеджмент проектов. Учеб. Курс.– СПб.: Питер, 2003.–636с.
5. Вигерс К.И. Разработка требований к ПО.– М.: Русская редакция Microsoft, 2004.–575с.
6. Кантор М. Управление программными проектами. Практическое руководство по разработке ПО. Пер.с англ.–М.:Вильямс, 2002.
7. Лаврищева К.М. Програмна інженерія. –Підручник.–К.: Академперіодика, 2008.–415с.
8. Лаврищева К.М. Методы программирования. Теория, инженерия, практика. – К.; Наукова Думка, 2006.–451с.
9. Леффингуэлл Д., Уидриг Д. Принципы работы с требованиями к программному обеспечению. Унифицированный подход. – М.: «Вильямс», 2002.- 448 с.
10. Соммервиль І. Інженерія програмного забезпечення. – М.; “Вильямс”, 2002. – 624 с../ home page (<http://www.software-engin.com>)

НАВЧАЛЬНЕ ВИДАННЯ

Білас Орест Євгенович
Мельник Наталія Богданівна

**СПЕЦИФІКАЦІЯ ВИМОГ
ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

**МЕТОДИЧНІ ВКАЗІВКИ
ДО ЛАБОРАТОРНИХ РОБІТ**
для студентів Інституту комп'ютерних наук та інформаційних технологій

*Редактор
Комп'ютерне верстання*