**Test Strategy**

**Step#1: Scope**

It defines parameters like

* Who will review the document?
* Who will approve this document?
* Testing activities carried out with timelines

**Step#2 Test Approach**

It defines

* Process of testing
* Testing levels
* Roles and responsibilities of each team member
* Types of Testing (Load testing, Security testing, Performance testing etc.)
* Testing approach & automation tool if applicable
* Adding new defects, re-testing,[Defect](https://www.guru99.com/the-unconventional-guide-to-defect-management.html)triage, [Regression Testing](https://www.guru99.com/regression-testing.html) and test sign off

**Step#3 Test Environment**

* Define number of requirement and setup required for each environment
* Define backup of test data and restore strategy

**Step#4 Testing Tools**

* Automation and Test management tools needed for test execution
* Figure out number of open-source as well as commercial tools required, and determine how many users are supported on it and plan accordingly

**Step#5 Release Control**

* Release management plan with appropriate version history that will make sure test execution for all modification in that release

**Step#6 Risk Analysis**

* List all risks that you can estimate
* Give a clear plan to mitigate the risks also a contingency plan

**Step#7 Review and Approvals**

* All these activities are reviewed and sign off by the business team, project management, development team, etc.
* Summary of review changes should be traced at the beginning of the document along with approved date, name, and commen

**Тест стратегія**

**Крок №1: сфера застосування, межі.**

**Він визначає такі параметри**

* **Хто перегляне документ?**
* **Хто схвалить цей документ?**
* **Тестування, визначене часовими сроками**

**Крок № 2 Тестовий підхід**

**Це визначає:**

**• Процес тестування**

**• Рівні тестування**

**• Типи тестування (Load testing, Security testing, Performance testing etc.))**

**• Тестовий підхід та інструмент автоматизації, якщо це застосовується**

**• Додавання нових дефектів, повторне тестування, виправлення дефектів, регресійне тестування та test sign off**

**Крок № 3 Тестування середовища**

**• Визначте кількість вимог і налаштувань, необхідних для кожного середовища**

**• Визначити резервну копію тестових даних(backup of test data )та відновити стратегію**

**Крок №4 Інструмент тестування**

**• Автоматизація та Тест менеджмент Інструменти необхідні для виконання тестування**

**• Визначте кількість відкритих джерел визначених комерційними інструментами,** **і визначити, скільки користувачів на ній підтримуються, і відповідно планувати**

**Крок №5 Контроль-Випуск**

**• Випуск плану управління з відповідною історією версій, що забезпечить виконання тесту для всіх модифікацій у цьому випуску**

**Крок №6 Аналіз ризиків**

**• Перерахуйте всі ризики, які ви можете оцінити**

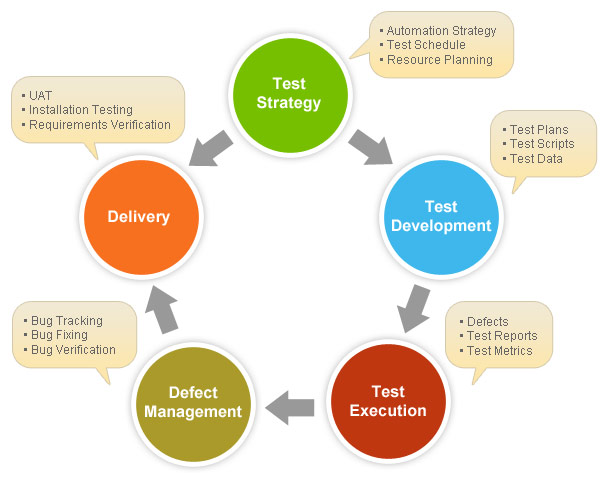
**• Надайте чіткий план пом'якшення ризиків, також план дій на випадок надзвичайних ситуацій**

**Крок №7 - огляд і схвалення**

**• Всі ці заходи розглядаються та підписуються бізнес-командою, керівництвом проекту, командою розробників тощо.**

**• Підсумки перегляду змін слід простежити на початку документа разом із затвердженою датою, іменем та коментарями**

### 1. Simple Tips

**1)**Include product background in test strategy document. In the first paragraph of your test strategy document answer – **Why stakeholders want to develop this project? This will help to understand and prioritize things quickly.**  
**2) List all important features you are going to test.** If you think some features are not part of this release then mention those features under “Features not to be tested” label.  
**3)**Write down the test approach for your project. **Clearly, mention what types of testing you are going to conduct?**I.e. Functional testing, UI testing, Integration testing, Load/Stress testing, Security testing etc.  
**4)** Answer questions like **How you are going to perform functional testing? Manual or automation testing? Are you going to execute all test cases from your test management tool?  
5)** **Which bug tracking tool you are going to use? What will be the process when you will find a new bug?**  
**6)** What are **your test entry and exit** criteria?  
**7)**How will **you track your testing** progress? What metrics are you going to use for tracking test completion?  
**8 )**Task distribution – **Define roles and responsibilities of each team member.**  
**9) What documents will you produce during** and after testing phase?  
**10)** **What all risk you see ?**

**Цели тестирования**  
Повысить вероятность того, что приложение, предназначенное для тестирования, будет работать правильно при любых обстоятельствах.  
Повысить вероятность того, что приложение, предназначенное для тестирования, будет соответствовать всем описанным требованиям.  
Предоставление актуальной информации о состоянии продукта на данный момент.

**Этапы тестирования:**

1. Анализ продукта  
2. Работа с требованиями  
3. Разработка стратегии тестирования и планирование процедур контроля качества  
4. Создание тестовой документации  
5. Тестирование прототипа  
6. Основное тестирование  
7. Стабилизация

8. Эксплуатация

**Тест план (Test Plan)** — это документ, описывающий весь объем работ по тестированию, начиная с описания объекта, стратегии, расписания, критериев начала и окончания тестирования, до необходимого в процессе работы оборудования, специальных знаний, а также оценки рисков с вариантами их разрешения.  
Отвечает на вопросы:  
Что надо тестировать?  
Что будете тестировать?  
Как будете тестировать?  
Когда будете тестировать?  
Критерии начала тестирования.  
Критерии окончания тестирования.

**Основные пункты тест плана**  
В стандарте перечислены пункты, из которых должен (пусть — может) состоять тест-план:   
a) Test plan identifier;   
b) Introduction;   
c) Test items;   
d) Features to be tested;   
e) Features not to be tested;   
f) Approach;   
g) Item pass/fail criteria;   
h) Suspension criteria and resumption requirements;   
i) Test deliverables;   
j) Testing tasks;   
k) Environmental needs;   
l) Responsibilities;   
m) Staffing and training needs;   
n) Schedule;   
o) Risks and contingencies;   
p) Approvals.

**(Components of the Test Plan document)**

 Test Plan id

 Introduction

 Test items

 Features to be tested

 Features not to be tested

 Test techniques

 Testing tasks

 Suspension criteria

 Features pass or fail criteria

 Test environment (Entry criteria, Exit criteria)

 Test deliverables

 Staff and training needs

 Responsibilities

 Schedule)

**Traceability matrix — Матрица соответствия требований** — это двумерная таблица, содержащая соответсвие функциональных требований (functional requirements) продукта и подготовленных тестовых сценариев (test cases). В заголовках колонок таблицы расположены требования, а в заголовках строк — тестовые сценарии. На пересечении — отметка, означающая, что требование текущей колонки покрыто тестовым сценарием текущей строки.  
Матрица соответсвия требований используется QA-инженерами для валидации покрытия продукта тестами. МСТ является неотъемлемой частью тест-плана.

**Тестовый случай (Test Case)** — это артефакт, описывающий совокупность шагов, конкретных условий и параметров, необходимых для проверки реализации тестируемой функции или её части.  
Пример:

**Action Expected and Actual Result Test Result**  
(passed/failed/blocked)  
Каждый тест кейс должен иметь 3 части:  
**PreConditions** Список действий, которые приводят систему к состоянию пригодному для проведения основной проверки. Либо список условий, выполнение которых говорит о том, что система находится в пригодном для проведения основного теста состояния.  
**Test Case Description** Список действий, переводящих систему из одного состояния в другое, для получения результата, на основании которого можно сделать вывод о удовлетворении реализации, поставленным требованиям  
**PostConditions** Список действий, переводящих систему в первоначальное состояние (состояние до проведения теста — initial state)  
Виды Тестовых Случаев:  
Тест кейсы разделяются по ожидаемому результату на позитивные и негативные:  
**• Позитивный** тест кейс использует только корректные данные и проверяет, что приложение правильно выполнило вызываемую функцию.  
**• Негативный тест** кейс оперирует как корректными так и некорректными данными (минимум 1 некорректный параметр) и ставит целью проверку исключительных ситуаций (срабатывание валидаторов), а также проверяет, что вызываемая приложением функция не выполняется при срабатывании валидатора.

**Баг Репорт (Bug Report)** — это документ, описывающий ситуацию или последовательность действий приведшую к некорректной работе объекта тестирования, с указанием причин и ожидаемого результата.  
Шапка  
Короткое описание (Summary) Короткое описание проблемы, явно указывающее на причину и тип ошибочной ситуации.  
Проект (Project) Название тестируемого проекта  
Компонент приложения (Component) Название части или функции тестируемого продукта  
Номер версии (Version) Версия на которой была найдена ошибка  
Серьезность (Severity) Наиболее распространена пятиуровневая система градации серьезности дефекта:  
• S1 Блокирующий (Blocker)  
• S2 Критический (Critical)  
• S3 Значительный (Major)  
• S4 Незначительный (Minor)  
• S5 Тривиальный (Trivial)  
Приоритет (Priority) Приоритет дефекта:  
• P1 Высокий (High)  
• P2 Средний (Medium)  
• P3 Низкий (Low)  
Статус (Status) Статус бага. Зависит от используемой процедуры и жизненного цикла бага (bug workflow and life cycle