# 12.1 Requirements

Name: _____ Class: _____ Date: _____ **Total: 24 marks**

**KEY WORDS** functional requirement • non-functional requirement

## Objective

Write testable requirements and trace them to tests.

## 1 Key knowledge and worked examples

### ■ The kinds

Functional: what the system does. Non-functional: how well. Use case: one actor goal, end to end ✓.

### ■ Worked example: the course list

FR-1 limit 5 (sixth rejected); FR-2 due = date + 21; FR-3 one reminder per overdue day; NFR-1 search p95 under 2 s ✓.

## 2 Practice

**2.1** "Search answers within 2 s" – which kind? [3]

**2.2** Give the testable rewrite of "the app should be easy to use". [3]

**2.3** Why is "use Redis" wrong inside a requirement? [3]

## 3 Exam-style questions

**3.1** The library adds a renewal rule: a member may renew a loan twice, each renewal extending the due date by 14 days from the current due date, only if no other member has reserved the copy.

Write FR-4 for the rule with a measurable clause. Write two acceptance tests (name, action, expected result). State one non-functional requirement the renewal feature might carry. Write the use-case sentence with its actor. [8]

**3.2** Trace FR-2 ("a borrow records the date and the due date 21 days later").

Give the test-first sequence: the failing test, the commit that closes it, and what the commit message must contain. Compute the due dates for borrows on 2026-02-10, 2028-02-10 and 2026-12-20. State what the suite checks about the requirement-test-code chain. [7]

## Solutions

**2.1**

- Non-functional – it constrains how well, not what ✓.

**2.2**

- E.g. “a new member completes first borrow in under 3 minutes without help (measured with 5 first-time users)“ ✓.

**2.3**

- It names a means, not a need – requirements state the need; solutions belong in design ✓.

**3.1**

- FR-4: a member may renew at most twice per loan; each renewal sets due = current due + 14 days; renewal is refused if another member holds a reservation on the copy ✓.
- Test 1 (renew ok): borrow with due D; renew; expect due $D+14$; renew again; expect $D+28$; a third renew is refused ✓.
- Test 2 (reservation blocks): reserve the copy to another member; renew; expect refusal with reason ✓.
- NFR: renewal responds within 1 s at p95 ✓.

**3.2**

- Sequence: red acceptance test (due = borrow date + 21, three date cases) → commit implementing the loan ledger’s date rule → message names FR-2 and the test ✓.
- 2026-02-10 + 21 = 2026-03-03; 2028-02-10 + 21 = 2028-03-02 (leap February carries 29 days); 2026-12-20 + 21 = 2027-01-10 ✓.
- The suite checks: the acceptance test exists, passes, and the closing commit links to FR-2 – the auditable chain ✓.

exercises.lol exercises.com.cn