

1 Multi-Currency Wallet Simulator

2 Risk Assessments – Start, Mid, End

3 This document describes the identified risks for the wallet simulator exam project and how
4 they are evaluated and tracked.

5 Each risk is rated using fixed scales for *Probability* and *Impact*, and a computed *Risk Factor* is
6 used to support prioritisation and test focus.

7 Risk Types

8 Risks are grouped into four types:

- 9 ▪ **Project risk** – Threats to planning, schedule, tools, or the ability to complete the work.
- 10 ▪ **Product risk** – Threats to the quality of the wallet simulator (functional and non-func-
11 tional).
- 12 ▪ **Testing risk** – Threats to the effectiveness or completeness of the testing activities.
- 13 ▪ **External risk** – Threats originating outside the project (e.g. external services, environ-
14 ment, hardware).

15 Each risk in the table is tagged with at least one *Risk Type*.

16 Probability Scale (Prob)

17 *Probability* expresses how likely it is that a risk event will occur during this project.

18 A discrete 1–5 scale is used:

- 19 ▪ **1 – Very unlikely:** Would be surprising; no similar issues seen in previous work.
- 20 ▪ **2 – Unlikely:** Could happen, but only under special circumstances.
- 21 ▪ **3 – Possible:** Realistic chance; may occur once during the project.
- 22 ▪ **4 – Likely:** Expected to occur unless actively mitigated.
- 23 ▪ **5 – Very likely:** Almost certain to occur given current plan and constraints.

24 Impact Scale

25 *Impact* expresses how severe the consequences are if the risk event occurs, considering both
26 project success and delivered product quality:

- 27 ▪ **1 – Very low:** Minor inconvenience; negligible effect on learning goals or result.
- 28 ▪ **2 – Low:** Small rework or delay; local quality issues, core goals still clearly met.

- 1 ▪ **3 – Medium:** Noticeable reduction in quality and/or scope; may affect some exam cri-
2 teria.
- 3 ▪ **4 – High:** Major rework or missing significant functionality; clearly harms perceived qual-
4 ity.
- 5 ▪ **5 – Very high:** Critical failure; core requirements, key tests, or deliverables cannot be
6 completed as intended.

7 **Risk Factor**

8 For each risk, a *Risk Factor* is calculated as:

9
$$\text{Risk Factor} = \text{Probability} \times \text{Impact}$$

10 This gives a value from 1 to 25.

11 Higher values indicate higher exposure and are used to prioritise mitigation actions and testing
12 effort.

13 **Status and Dates**

14 Each risk in the table includes:

- 15 ▪ **Status** – current handling:
 - 16 ○ **Open** – identified and relevant; actions still needed.
 - 17 ○ **Monitoring** – mitigation in place; risk is being observed.
 - 18 ○ **Accepted** – consciously accepted without further mitigation.
 - 19 ○ **Closed** – no longer relevant or fully resolved.
- 20 ▪ **Status Date** – when the status and ratings were last updated.
- 21 ▪ **Follow-up Date** – when the risk is planned to be reassessed.

22 For this exam project, the risk assessment is performed at three points:

- 23 ▪ **Initial** – after SRS v2 is stabilised.
- 24 ▪ **Mid** – before API testing is started.
- 25 ▪ **End** – before final deliverables are completed.

26 Each version of the risk table represents a snapshot at one of these assessment points.

27 See next pages for Risk Tables.

1 Risk Assessment – Risk Table (Start)

2 (Snapshot date: 22-11-2025 – ordered by Risk Factor, Impact, ID)

ID	Risk Type	Risk Name	Risk Description	Prob (1–5)	Impact (1–5)	Risk Factor $P \times I$	Mitigation	Responsible	Status	Follow-up Date
R7	Project	Insufficient time for tests and documentation	Effort for designing, implementing, and documenting all required tests and reports may be underestimated, leading to incomplete deliverables.	4	5	20	Create a simple schedule that reserves time per test type, prioritise high-risk areas, and de-scope low-impact extras if needed.	Project Manager	Open	Mid assessment (before API testing)
R1	Product	Incorrect rounding in currency exchange	Rounding for cross-currency exchanges may not follow the specified “round half up to 2 decimals”, causing incorrect wallet balances.	3	5	15	Implement a single rounding utility used by all exchange operations and cover it with boundary-focused unit tests and black-box test cases from the SRS.	Backend Developer	Open	Mid assessment (before API testing)
R2	Product	Wallet status and balance rules violated	Wallet status transitions or rules for operations on non-active wallets may be implemented incorrectly, or successful operations may allow negative balances.	3	5	15	Derive a clear state diagram from the SRS and design state-based tests to cover all allowed and forbidden transitions and operations per status.	Test Architect	Open	Mid assessment (before API testing)
R3	Product	Transaction recording and	Transactions may not be recorded as specified, or balances and transaction records may become inconsistent in error scenarios.	3	4	12	Define transaction persistence rules explicitly and verify them with integration tests that assert both DB state and	Backend Developer	Open	Mid assessment (before API testing)

		atomicity incorrect					balances for success and failure cases.			
R5	External / Product	External exchange-rate API availability and reliability	The external exchange-rate API may be unavailable, slow, or change format, causing exchange operations to fail unexpectedly.	3	4	12	Implement timeouts, input validation of API responses, and clear fallback behaviour (fail exchange without state change) plus integration tests for these cases.	Backend Developer	Open	Mid assessment (before API testing)
R8	Testing	Test design not aligned with high-risk areas and SRS	Test cases may focus on easy paths rather than the most critical rules (rounding, status rules, invalid input), leaving important gaps.	3	4	12	Use the Risk Assessment and SRS to prioritise test design and ensure that the highest-risk requirements are covered by explicit black-box and unit tests.	Test Engineer	Open	Mid assessment (before API testing)
R11	Project / Product	Misalignment between SRS, Risk Assessment, and tests	Implementation and tests may drift from SRS v2 and from the assumptions in this Risk Assessment, weakening traceability and justification.	3	4	12	Regularly cross-check SRS, Risk Assessment, and test cases, updating documents and tests when scope or interpretation changes.	Test Manager	Open	Mid and End assessments
R4	Product / Testing	Inconsistent validation and error responses	Invalid input handling, HTTP status codes, and error payloads may differ between endpoints, reducing API predictability and test reliability.	3	3	9	Define a small error-response contract (status codes + error codes) and cover it with API tests for typical invalid-input and server-error scenarios.	Test Engineer	Open	Mid assessment (before API testing)

R6	Project / External	Tooling and environment issues delaying testing	Problems with tool setup (e.g. Playwright, JMeter) or environment configuration may consume time and block test execution.	3	3	9	Set up and smoke-test all required tools and environments early and document setup steps for repeatable execution.	DevOps Engineer	Open	Mid assessment (before API testing)
R9	Testing	Insufficient API and end-to-end coverage for error flows	API and UI tests may not thoroughly cover error situations and realistic flows, reducing confidence in end-to-end behaviour.	3	3	9	Design API and UI test scenarios that explicitly include invalid inputs, state violations, and external API failures across end-to-end flows.	Test Engineer	Open	Mid assessment (before API testing)
R10	Testing	Performance behaviour not meaningfully validated	Performance and stress tests may be too shallow or missing, giving little evidence of response times under basic concurrent use.	2	3	6	Define a small but realistic performance scenario for key endpoints and run it with a load tool, documenting response times and any bottlenecks.	Performance Test Engineer	Open	End assessment (before final deliverables)

1 Risk Assessment – Risk Matrix (Start)

2 (Snapshot date: 22-11-2025)

Impact \ Probability	1 – Very low	2 – Low	3 – Moderate	4 – High	5 – Very high
5 – Very Likely					
4 – Likely					R7
3 – Possible			R4, R6, R9	R3, R5, R8, R11	R1, R2
2 – Unlikely			R10		
1 – Very Unlikely					

1 Risk Assessment – Risk Table (Mid)

2 (Snapshot date: 06-12-2025 – ordered by Risk Factor, Impact, ID)

ID	Risk Type	Risk Name	Risk Description	Prob (1-5)	Impact (1-5)	Risk Factor $P \times I$	Mitigation	Responsible	Status	Follow-up Date
R7	Project	Insufficient time for tests and documentation	Effort for designing, implementing, and documenting all required tests and reports may be underestimated, leading to incomplete deliverables.	4	5	20	Create a simple schedule that reserves time per test type, prioritise high-risk areas, and de-scope low-impact extras if needed.	Project Manager	Open	End assessment
R1	Product	Incorrect rounding in currency exchange	Rounding for cross-currency exchanges may not follow “round half up to 2 decimals”, causing incorrect wallet balances.	3	5	15	Implement a single rounding utility used by all exchange operations and cover it with boundary-focused unit tests and black-box test cases from the SRS.	Backend Developer	Open	End assessment
R2	Product	Wallet status and balance rules violated	Wallet status transitions or rules for operations on non-active wallets may be implemented incorrectly, or successful operations may allow negative balances.	3	5	15	Derive a clear state diagram from the SRS and design state-based tests to cover allowed/forbidden transitions and operations per status.	Test Architect	Open	End assessment
R3	Product	Transaction recording and atomicity incorrect	Transactions may not be recorded as specified, or balances and transaction records may become inconsistent in error scenarios.	3	4	12	Define transaction persistence rules explicitly and verify with integration tests asserting both DB state and balances for success/failure.	Backend Developer	Open	End assessment

R5	External / Product	External exchange-rate API availability and reliability	External FX API may be unavailable/slow/change format, causing exchange failures.	3	4	12	Implement timeouts, response validation, and clear fallback behaviour (fail without state change) plus tests for these cases.	Backend Developer	Open	End assessment
R8	Testing	Test design not aligned with high-risk areas and SRS	Tests may focus on easy paths rather than critical rules (rounding, status, invalid input), leaving gaps.	3	4	12	Use Risk Assessment + SRS to prioritise tests so highest-risk requirements are explicitly covered.	Test Engineer	Open	End assessment
R11	Project / Product	Misalignment between SRS, Risk Assessment, and tests	Implementation/tests may drift from SRS and assumptions, weakening traceability.	3	4	12	Regularly cross-check SRS, Risk Assessment, and tests; update docs/tests when scope or interpretation changes.	Test Manager	Open	End assessment
R4	Product / Testing	Inconsistent validation and error responses	Invalid input handling, HTTP status codes, and error payloads may differ between endpoints.	3	3	9	Define a small error-response contract and cover with API tests for typical invalid-input and server-error scenarios.	Test Engineer	Open	End assessment
R6	Project / External	Tooling and environment issues	Tool setup and environment config may consume time and block test execution.	3	3	9	Set up and smoke-test tools early and document setup steps for repeatable execution.	DevOps Engineer	Monitoring	End assessment

		delaying testing								
R9	Testing	Insufficient API and end-to-end coverage for error flows	API/UI tests may not cover error situations and realistic flows well enough.	3	3	9	Design API/UI scenarios including invalid inputs, state violations, and external API failures across flows.	Test Engineer	Open	End assessment
R10	Testing	Performance behaviour not meaningfully validated	Performance tests may be too shallow/missing, giving little evidence under concurrent use.	2	3	6	Define a small realistic performance scenario for key endpoints and run it with a load tool, documenting response times/bottlenecks.	Performance Test Engineer	Open	End assessment

1 What changed since Start (Mid)

- 2 ▪ The list of risks is still the same, but some uncertainty is reduced as the project becomes clearer.
- 3 ▪ Tooling risk (R6) is moved from “Open” to “Monitoring” because the environment is more stable than at Start.
- 4 ▪ The top risks are still time pressure (R7) and correctness of key rules (R1, R2).
- 5 ▪ Most risks stay “Open” because Mid is intentionally before the heavier test work (API/E2E/performance).

1 **Risk Assessment – Risk Matrix (Mid)**

2 *(Snapshot date: 06-12-2025)*

Impact \ Probability	1 – Very low	2 – Low	3 – Moderate	4 – High	5 – Very high
5 – Very Likely					
4 – Likely					R7
3 – Possible			R4, R6, R9	R3, R5, R8, R11	R1, R2
2 – Unlikely			R10		
1 – Very Unlikely					

1 Risk Assessment – Risk Table (End)

2 (Snapshot date: 13-12-2025 – ordered by Risk Factor, Impact, ID)

ID	Risk Type	Risk Name	Risk Description	Prob (1-5)	Impact (1-5)	Risk Factor $P \times I$	Mitigation	Responsible	Status	Follow-up Date
R7	Project	Insufficient time for tests and documentation	Risk of running out of time for final packaging and evidence.	3	4	12	Reserve fixed time blocks for final evidence and keep scope tight near the end.	Project Manager	Closed	N/A (final)
R1	Product	Incorrect rounding in currency exchange	Incorrect balances if rounding rules are wrong in edge cases.	2	5	10	Keep a single rounding utility and test boundary values.	Backend Developer	Closed	N/A (final)
R2	Product	Wallet status and balance rules violated	State rules could be wrong in edge transitions or blocked operations.	2	5	10	Keep state rules explicit and test allowed/forbidden operations per state.	Test Architect	Closed	N/A (final)
R12	External / Testing	Performance results misleading due to stubbed FX	Performance tests may look better with FX stub than with real external dependency.	3	3	9	State test assumption clearly (FX stubbed) and treat results as internal baseline, not real-world SLA.	Performance Test Engineer	Accepted	N/A (final)

R3	Product	Transac-tion re-cording and ato-micity in-correct	Inconsistencies could ap-pear under error paths or concurrency.	2	4	8	Keep transaction rules ex-plicit and verify success/fail-ure paths across layers.	Backend Developer	Closed	N/A (final)
R5	Exter-nal / Product	External exchange-rate API availability and reliability	Real FX dependency can fail or change behaviour.	2	4	8	Keep timeouts/validation and fail without state change; keep stub available for test reliability.	Backend Developer	Closed	N/A (final)
R11	Project / Prod-uct	Misalign-ment be-tween SRS, Risk Assess-ment, and tests	Small drift can happen dur-ing last changes.	2	4	8	Do a final cross-check: SRS ↔ tests ↔ API contracts.	Test Man-ager	Closed	N/A (final)
R4	Product / Test-ing	Incon-sistent val-idation and error responses	Some invalid inputs might still behave differently across endpoints.	2	3	6	Keep one shared parsing/val-idation approach and verify key 400/422/502 contracts.	Test Engi-neer	Closed	N/A (final)
R8	Testing	Test de-sign not aligned with high-	Remaining gaps might be in corner cases rather than main flows.	2	3	6	Keep prioritising round-ing/state/error contracts over cosmetic coverage.	Test Engi-neer	Closed	N/A (final)

		risk areas and SRS								
R9	Testing	Insufficient API and end-to-end coverage for error flows	E2E/API might still miss rare combinations.	2	3	6	Keep a small set of error-flow tests that match the SRS contracts.	Test Engineer	Closed	N/A (final)
R10	Testing	Performance behaviour not meaningfully validated	Risk reduced once a basic load/stress/spike baseline exists, but results are still limited.	2	3	6	Keep results as evidence of baseline behaviour, not production guarantees.	Performance Test Engineer	Closed	N/A (final)
R6	Project / External	Tooling and environment issues delaying testing	Tooling issues are now less likely late stage.	1	3	3	Keep instructions + configs versioned and repeatable.	DevOps Engineer	Closed	N/A (final)

1 What changed since Mid (End)

- 2 ▪ Probability is reduced for several items because fewer unknowns remain near the end (especially R6).
- 3 ▪ The biggest remaining risk is still time pressure (R7), but impact is lower than at Start because the scope is more defined.
- 4 ▪ External dependency risks (R5) remain but are easier to manage when testing uses a controlled FX setup.
- 5 ▪ A new risk is added (R12): performance results are useful as a baseline but can't be treated as "real internet" performance when FX is stubbed.

1 **Risk Assessment – Risk Matrix (End)**

2 (*Snapshot date: 13-12-2025*)

Impact \ Probability	1 – Very low	2 – Low	3 – Moderate	4 – High	5 – Very high
5 – Very Likely					
4 – Likely					
3 – Possible			R12	R7	
2 – Unlikely			R4, R8, R9, R10	R3, R5, R11	R1, R2
1 – Very Unlikely			R6		