

## **Test Plan**

### **Monefy Application**

**Objective:** ensure that the software under test is bug free before release

**Deadline:** Q3

**People resources:**

2 QA Engineers (Junior QA Engineer, QA Automation Engineer), QA Manager

**Dependency & Risks:**

- environment depends on DevOps team (prepare and provide environment specification in advance)
- team member lack the required skills for mobile testing (training course to skill up the team)
- Project schedule is too tight (set Priority for each of test activities)

**Types of tests:**

- Development team: Unit tests
- QA team: Manual exploratory and end-to-end testing, GUI testing, UI automation testing, API automation testing, Functional testing, Performance testing, Compatibility testing, Usability testing, Regression testing
- Out of scope: Security testing, Database testing, external interfaces

**Tools and technologies:** Appium (Python), Android SDK, JDK, Robot framework - API testing (Python with seleniumlibrary), Gatling - performance/load testing (Scala DSL), Postman - manual API testing, Jenkins, Maven, IntelliJ IDEA

**Test artifacts:** test plan (QA Manager), test cases/regression test suite/smoke test suite/smoke checklist/regression checklist (QA Engineers), test reports/release notes (QA Manager)

**Milestones:**

- creation of test plan, test strategy documentation and review with a team
- test environment specification and setup
- define the development/deployment workflow
- manual exploratory testing and defining test cases for automation, documentation creation
- prepare test data
- start UI automation testing with continuous increment
- implementation of API automated tests
- defining of smoke, regression test suites
- integration of existed test suites to the deployment pipeline

- perform load testing and implementation of automated performance tests
- maintenance of existed tests and increasing the test coverage
- prepare test report and release notes

#### **Test Criteria:**

1. Critical suspension criteria: 40% of test cases failed (suspend testing until the development team fixes all the failed cases)
2. Exit criteria: 95% off all critical test cases are passed

**Device coverage:** iOS, Android devices with different OS versions, size of the screen, browsers (Windows Phone is out of scope) based on data from analytics

#### **P0: Functional test cases checklist:**

1. Installation
  - Install on Android (supported versions)
  - Install on iOS (supported versions)
2. Application launch
  - Validate with different preconditions (memory/battery..)
3. Tutorial validation
  - Validate every step of tutorial for new use
  - Validate that tutorial is not shown for not a new user
4. Empty state
  - Validate empty state (no transactions)
  - Validate that balance = 0
  - Validate that transactions list is empty
5. Navigation menu (accounts and settings)
  - Check that menu could be opened and closed
  - Validated all webelements
  - Validate interaction between menus
6. Header
  - Validate that it is possible to open accounts menu
  - Validate that it is possible to open settings menu
  - Validate that it is possible to open "transfer" step
  - UI validation
7. Add new transfer functionality
  - Validate that user is able to add new transfer "from header"
  - Validate that it is possible to add transfer just between existed account
  - Check input fields validation
  - Validate that it is possible to input amount
  - Validate that it is possible input note
  - Check that just existed enabled accounts are available in dropdown
  - Validate UI elements
  - Validate that it is possible to open calendar

- Validate that it is possible to set transfer date
  - Validate that it is not possible to transfer money within one account
  - Validate that it is possible to navigate back to main page
8. Add new Expense
- Validate that user is able to add new Expense from main screen
  - Check validation for input fields
  - Validate UI elements
  - Validate that it is possible to assign just existed category
  - It should be possible to add note
  - It should be possible to change transaction date
  - It should be possible to change transaction account
  - It should be possible to calculate transactions in calculator
  - Validate navigation after transaction was added
  - Validate that it is possible to move back to main page
  - Validate that transaction is displayed in transactions (general/per category/per account)
9. Add new Income
- Validate that user is able to add new Income from main screen
10. Add new Expense for category
- Validate that user is able to add new Expense for category (for each category)
  - Validate that proper category is assigned to transaction
11. Check Balance
- Validate calculation of Balance
12. Check List of transactions and transactions details
- Validate list of transactions and transactions details
13. Edit transaction functionality
- Validate that user is able to edit transaction
14. Delete transaction functionality
- Validate that user is able to delete transaction
15. Change account functionality
- Validate that user is able to change account
16. Display balances/transaction list per account
- Validate Balance calculation / list of transactions per account
17. Timerange functionality
- Validate that it is possible to set time range for transactions and balance calculation
18. Edit category details
- Validate that it is possible to edit category details
19. Delete category
- Validate that it is possible to delete category
20. Enable category
- Validate that it is possible to enable category
21. Merge category

- Validate that it is possible to merge category
- 22. Add new account
  - Validate that it is possible to add new account
- 23. Edit account
  - Validate that it is possible to edit account
- 24. Delete account
  - Validate that it is possible to delete account
- 25. Add currency rate
  - Validate that it is possible add currency rate
- 26. Enable account
  - Validate that it is possible to enable account
- 27. General settings
  - Validate that it is possible to:
    1. Clear data
    2. Restore data
    3. Create data backup
    4. Google drive sync
    5. Dropbox sync
    6. Export to file
    7. Add review
    8. Check About Monefy and Privacy Policy info
    9. Passcode protection functionality: passcode and fingerprint
    10. Setup first day of month
    11. Setup first day of week
    12. Chose main currency
    13. Chose application language
    14. Change theme
    15. Set budgets
- 28. Add new categories for Income and Expenses
- 29. Buy pro account
- 30. Subscription functionality
- 31. Calculation of Balance based on account/time range/category
- 32. Uninstallation

**P1: Test checklist:**

1. Validate that application works on all supported platforms:
  - iOS last version, the most popular versions (P0)
  - Android last version, two the most popular versions (P0)
  - Compatibility with the most popular browsers per OS (P0)
  - All other OSs and browsers from device coverage matrix

**P2: Test testing:**

- Usability testing

- Load testing
- Stress testing