Notes Editor

Prepared by:

Kudriavtsev Oleksii

12.01.2024

TABLE OF CONTENTS

1.0 INTRODUCTION

2.0 OBJECTIVES AND TASKS

2.1 Objectives

2.2 Tasks

3.0 SCOPE

4.0 Testing Strategy

4.1 Exploratory Testing

5.0 Exit Criteria for Tests

6.0 Hardware Requirements

7.0 FEATURES TO BE TESTED

8.0 SCHEDULES

9.0 TOOLS

**1.0 INTRODUCTION**

Tested note editor on Android 13

**2.0 OBJECTIVES AND TASKS**

2.1 Objectives

Check the functionality of the note editor mobile application on the Android platform version 4 and higher.

2.2 Tasks

The testing tasks should cover various aspects of the application and ensure its stable and effective operation under different usage conditions.

**3.0 SCOPE**

Testing of Basic Functions:

* Creation, Editing, and Deletion of Notes:
  + Testing the ability to create, edit, and delete notes.
  + Ensuring that changes made to notes are properly saved.
* Definition and Modification of Priorities and Categories:
  + Testing the functionality to define and modify priorities and categories of notes.
  + Verifying that changes to priorities and categories are reflected accurately.
* Adding Media Files (Photos, Voice Notes):
  + Testing the capability to attach media files, such as photos and voice notes, to notes.
  + Ensuring proper display and accessibility of attached media files.

Testing of Interfaces:

* Usability Testing:
  + Checking the convenience of using basic interface elements by the user.
  + Identifying potential improvements regarding user navigation.
* Interaction Testing with Calendar and Other Apps:
  + Testing the interaction of notes with the calendar and other applications.
  + Verifying how the application integrates with other services.

Performance Testing:

* Measurement of Note Creation Time:
  + Measuring the time taken to create a varying number of notes.
  + Assessing the application's responsiveness to note creation.
* Testing Application Response to a Large Number of Saved Notes:
  + Assessing the application's performance when dealing with a significant number of saved notes.
  + Identifying any impact on the application's speed and responsiveness.

Testing in Offline Mode:

* + Testing Application Functionality without Internet Connection:
  + Verifying the application's ability to function without an internet connection.
  + Ensuring that essential features remain accessible even in the absence of an internet connection.

**4.0 Testing Strategy**

4.1 Exploratory Testing

Definition:

Exploratory testing is a method where a tester examines the functionality of a system without predefined test scenarios. The main idea is to give the tester freedom to explore various aspects of the software product to uncover defects that might go unnoticed during other types of testing. This method can effectively save time as it allows direct identification of issues and defects that may be challenging to predict or describe beforehand, making it efficient compared to traditional testing methods.

Participants:

Kudriavtsev Oleksii -Tester

Methodology:

Understanding the Product:

* Familiarization with the requirements and documentation of the product.
* Meetings with developers and business analysts for a better understanding of functionality.

Functional Testing:

* Selection of specific functional areas for investigation.
* Testing different input variations and value ranges to verify functionality.

Navigation and Interface:

* Analysis of the user interface to determine logic and usability.
* Verification of the system's response to various interactions with interface elements.

Stress Testing:

* Creation of stress testing scenarios to assess system endurance.
* Increasing the load to identify functional limits and determine recovery time.

Documentation of Results:

* Recording all identified defects and anomalies.
* Generating a report, including descriptions of paths taken, identified issues, and recommendations for resolution.

**5.0 Exit Criteria for Tests:**

To conclude the testing process and recognize it as successful, the following criteria must be met:

* Number of Tests: Complete the execution of 30 test cases in accordance with the testing plan.

**6.0 HARDWARE REQUIREMENTS**

6.1 Computers

For efficient testing of the Android mobile application (version 4+) on the computer, the following computer specifications are recommended:

* Processor: Minimum Intel Core i5 or equivalent.
* RAM: Minimum 8 GB.
* Operating System: Windows 10 or macOS Catalina.
* Android Studio: Installed and configured for emulating Android devices.

6.2 Android Emulation

To emulate mobile devices on the computer, an Android emulator should be utilized with the following settings:

* Android Version: Minimum Android 4.0 (Ice Cream Sandwich) or higher.
* Screen Resolution: Emulators should be configured with diverse screen resolutions to assess adaptive design.

**7.0 FEATURES TO BE TESTED**

Regression tests:

Description: Verify the stability and functionality of test cases numbered 4, 5, 11, 12, 18, 22, 24, 26, 27 after updating the software.

Expected Result: All test cases should pass without errors.

Offline Test Cases:

Description: Reproduce test scenarios without Internet access (disable internet on the phone, turn off the router, sudden loss of internet during operation in the editor), checking the interaction with the program and its functionality.

Expected Result: The program should function correctly, even in the absence of an internet connection.

Compatibility Testing on Various Android Versions

Description:Conduct tests on different versions of the Android operating system to verify the compatibility of the application and ensure its stable operation.  
Expected result:All tests should be successfully executed without encountering errors or abnormalities on all tested versions of Android.

**8.0 SCHEDULES - Major Deliverables**

The major deliverables for the testing phase include the following documents:

* Test Plan
* Test Cases
* Test Summary Reports

9.0 TOOLS

* Test Case Management: Excel (Manual)
* Bug Tracking: Excel
* Automation: N/A (Manual Testing)