ANTI-VIRUS APPLICATION FOR ANDROID

PRESENTED BY:

- AFRA SADAT (002)
- TOOBA KHALIQUE (031)

TABLE OF CONTENTS

INTRODUCTION

Anti-Virus Software

Software that is created specifically to help detect, prevent and remove malware.

Antivirus is a kind of software used to prevent, scan, detect and delete viruses from a computer. Once installed, most antivirus software runs automatically in the background to provide real-time protection against virus attacks.

Nowadays anti-virus software applications have become essential to the everyday personal android user. These modern tools for android malware scanning use a mixture of algorithms to detect and prevent malicious software from causing damage. This project is about building an anti-virus tool for scanning file systems and detecting malware.

Advantages

Comprehensive virus protection programs help protect your files and hardware from malware such as worms, Trojan horses and spyware, and may also offer additional protection such as customizable firewalls and website blocking.

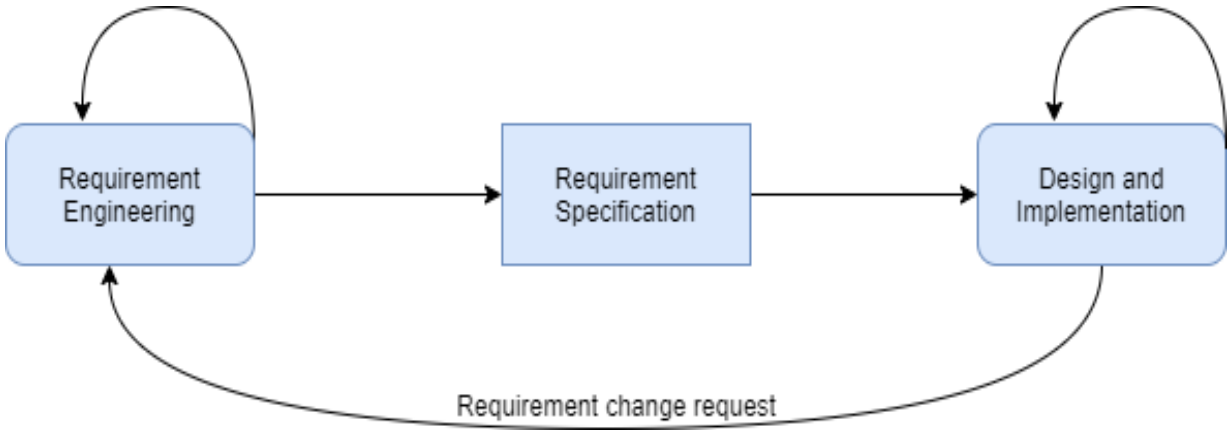
Working

Antivirus software begins operating by checking your computer programs and files against a database of known types of malware. Since new viruses are constantly created and distributed by hackers, it will also scan computers for the possibility of new or unknown type of malware threats.

Model Used In Project

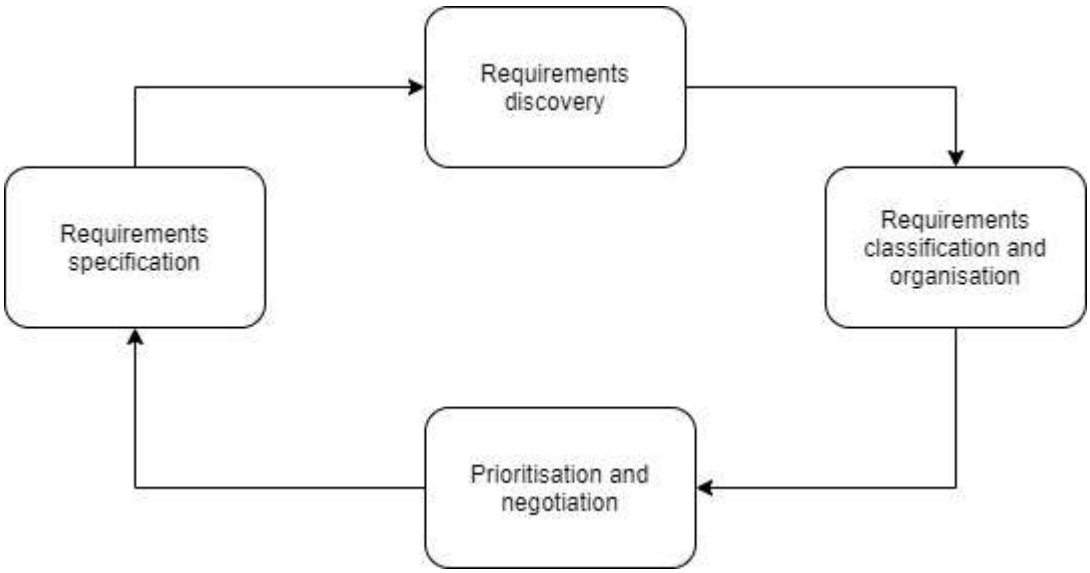
We use agile model for the development of our antivirus application. Agile development model is Incremental model. Software is developed in incremental, rapid cycles. By testing each release, we can ensure that the software quality is maintained. Customer satisfaction is high due to frequent communication after each processed cycle and changing requirements are easily manages into the project. Moreover, the customer is able to view the development in project and contact at any point to the developer about changings in his demands. Even late changes in requirements are welcomed.

When new changes are needed to be implemented, the freedom agile gives to change is very important. New changes can be implemented at very little cost because of the frequency of new increments that are produced. To implement a new feature the developers, need to lose only the work of a few days, or even only hours, to roll back and implement it. Unlike the waterfall model in agile model very limited planning is required to get started with the project. Agile assumes that the end users' needs are ever changing in a dynamic business. Changes can be discussed and features can be newly effected or removed based on feedback. This effectively gives the customer the finished system they want or need.
The cycle of Agile development is



REQUIREMENT ENGINEERING

The requirement engineering cycle consists of following steps that we will repeat for each phase.



Requirement Discovery

For requirement elicitation a meeting was held between the stakeholders for requirement elicitation process.
The stakeholders include:

- Project Managers
- Developer
- Customers
- Sponsors

After interviewing the customers for their requirements and discussing the cost of the project with the sponsors, we have come to conclude that our customer wants a system that is able to scan, detect and remove malware and viruses. Additional features should include real-time scanning, choice for removing selected malicious files**,** protection for apps**,** app lock**,** power saving mode, app hide option, junk files detection and deletion, compatibility, independent of network availability and updates will be available for each new version of the app. Furthermore, it should be available 24/7. The project should be low cost. Security & Performance of the system should be effective. System should be flexible for changes and modification.

Requirement Classification

After collecting requirements from customer, for next step first we will classify and organize our requirements into defined groups.

Functional Requirements:

- ***Scanning***
  The user should be able to scan the entire android system of his phone for viruses and malware.
- ***Detection***
  The app should be able to detect any malware or virus through static and dynamic file analysis.
- ***Deletion***
  The user should be able to delete the malware or virus affected files by clicking on them.
- ***Real-time Scanning***
  Whenever a file will be installed or downloaded, system should automatically scan it for malware and viruses and notify the user if the file is affected and prevent it from accessing until granted permission by user.
- ***App Protection***
  This feature should allow user to prevent any specific app from getting attacked by data breaching, malwares or other viruses. This app will be scanned for viruses every 24 hours and whenever it will be used. Furthermore, in case of any malware or virus found it will be deleted automatically without permission and user will only be notified.
- ***App Lock***
  This feature should allow user to lock any specific app with password to limit its access. Each time someone will try to open it, password notification should be displayed and only upon correct password app will proceed.
- ***Power Saving Mode***
  This mode when enabled should close all un-necessary background running apps, reduce brightness and notify user whenever he will try to open a heavy app that will consume more RAM.
- ***App Hide Mode***
  This mode when enabled should hide the selected app and then it can only be moved back to home screen by choosing the app again and clicking the unhide option from our anti-virus app.
- ***Junk Files Detection & Deletion***
  This feature should scan the system for junk files, show user the space occupied by junk files and then delete them when requested by user.

- ***Notify for Updates available***
  This feature should be able to notify the user whenever a new version of software is launched.

Non-Functional Requirement:

- ***Compatibility***
  This is a daily life application and a need of every android mobile. This antivirus application should be available for every version of android.
- ***Independent of Network availability***
  This application should work without the need of internet. Once it is downloaded, it should be accessible without internet.
- ***Availability***
  The application should have no time limit. It should be available 24/7 whenever needed.
- ***Security***
  Safety and security are a main concern for any user. The application must guarantee that personal data of user is safe and secure and that private data is encrypted. Additionally, the system ought to be protected from any type of attack of system itself.
- ***High Performance***
  The performance of the application should be efficient. The ultimate goal of performance monitoring is to supply end users with a top-quality end-user experience.
- ***Flexibility***
  The application should be flexible. Time to time user needs additional features. Changes should be easily implemented at any time.
- ***Cost-effective***
  This application should be developed in customer desired cost. The features can be minimized but the cost will not increase.

Domain Requirements:

- ***Choice for removing selected malicious files***
  The system should give user choice after showing the affected files that which files user wants to delete and which files he wants to keep.

Prioritization & Negotiation

After collecting and classifying our requirements, now we will prioritize our requirements for their need in our project.

Functional requirements are the features we want to add in our system. Scanning, Detection, Deletion and Real-time Scanning are the major features which cannot be eliminated at any cost and are must to be added. App Protection, Junk Files Detection & Deletion and Notification for Updates available are the additional features that if not added will not affect the main features but if added will make our system multi-purpose. App Lock, Power Saving Mode and App Hide Mode are also additional features but differ from the nature from our main system so these three are the least ones in our priority.

After prioritizing, now we will negotiate with our clients to remove some of the features as the cost of the project is not fit for all of the user demands.

We will give user three options of which two requirements he wants to remove as our client was sure that the cost is not negotiable so we have to deduct some features.

Power saving mode is un-necessary feature as it differs from the nature of our system and mostly android phones have this feature as built-in.

App-Lock feature doesn't differ from the nature of our system but many apps are found on web solely for this purpose and this will increase project cost and size as this feature is heavy**.**

App Hide Mode feature will increase the cost and size of our project and the main need our end-users is protection from viruses so this feature can be added if cost is to be increased.

After consulting our client two requirements has been deducted from our system. *App Lock* and *Power Saving Mode* are no longer included in user requirements.

System Requirement Specification

Stake Holder	Requirement	Classification of Requirement	Description
Customer	Scanning for viruses & malware	Functional	Scanning each and every file in the whole android system to search for any virus or malware.
	Detecting Infected Files	Functional	Detecting the infected files and showing them to user.
	Choosing Desired Files	Domain	Selecting desired files from the list of detected files.
	Deleting Infected Files	Functional	Deleting All or Selected Infected Files on user's command.

	Real-time Scanning	Functional	Automatically scan for viruses and malware whenever a file or app is installed.
	App Protection	Functional	Protection for any specific app from data breaching, malwares or other viruses.
	App Hide Mode	Functional	Hide any specific app from Home screen.
	Junk Files Detection & Deletion	Functional	Detect and delete Junk files from the entire system.
	Notify for Updates	Functional	Notify user if there are updates available for the new version of product.
	Compatibility	Non-Functional	Product should be compatible with every kind of android system.
	Independent of Network Availability	Non-Functional	System should work even without Internet Availability.
	Availability	Non-Functional	System should be available all time.
	Security	Non-Functional	No data breaching and protection of user privacy.
	High Performance	Non-Functional	Effective Performance in handling tasks.
	Flexibility	Non-Functional	Easy to modify and change.
Sponsor	Cost-effective	Non-Functional	Product cost should not increase from the decided cost.

ARCHITECTURAL MODEL

ARCHITECTURAL DESIGN

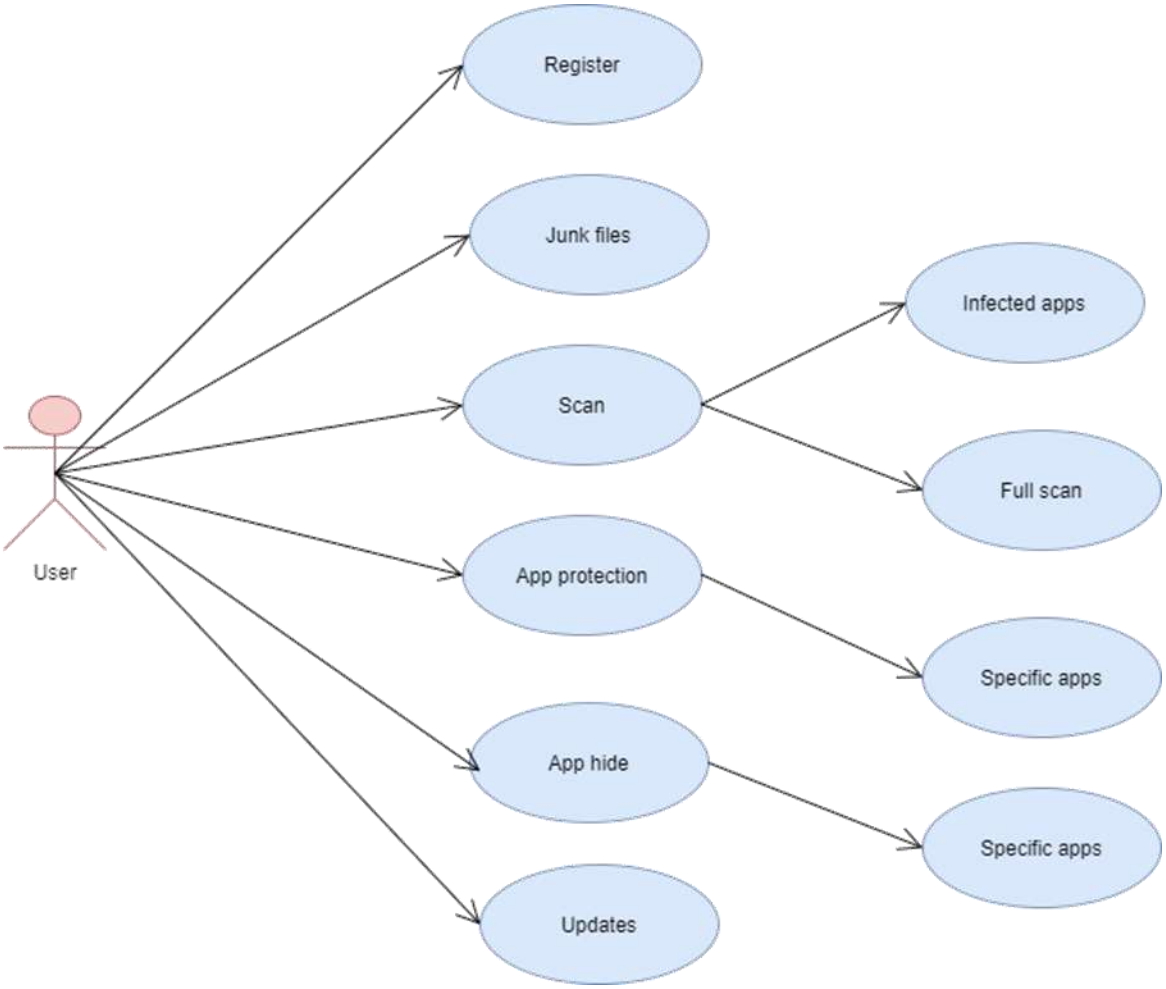
Architectural Design	Description
Registration	When the user installs the app, first he is asked to register. For registration the app asks user to enter Name, E-mail and Password. On entering that and clicking enter, a mail is sent to the written email to validate account. The mail consists of a link and a code. Either the user enters the code in the screen showing on the app after entering data or he can click on that code to verify his account.
Menu	After registration, a screen appears containing different options. The user is given options of ***Running a Quick Scan, Protect an app, App Hide Mode*** and ***Remove Junk Files.***
Running a Quick Scan	In this the entire system is scanned for viruses and malware within almost 3 to 4 minutes. Then the total number of infected files, their name and location is showed in a list. The user has the option to either select all files and delete them or he can select some from the list and delete them.
Protect an App	In this the user can select the app from the list shown on screen. The app selected is protected against malware and viruses. In the same way, if the user wants to remove protection from app then go to the Protect an App menu, choose protected app options. This will show the list of apps currently protected. Remove the app from there to remove protection from it.
App Hide Mode	In this the user can select the app from the list shown on screen. The app selected gets hidden from Home Screen only. In the same way, if the user wants to show app on Home screen, then go to the App Hide Mode menu, choose Hidden Apps options. This will show the list of apps currently Hidden. Unhide the app from there to show it on Home screen.
Remove Junk Files	This menu gives user the option to run a scan to find junk files in system. Once the junk files are detected which takes

	almost 2 to 3 minutes, then the user is shown all the junk files sorted category wise. The space taken by each category such as photos, music, videos etc. is shown separately. Beneath this the space taken by junk files is shown app wise that how much junk files are found in each app. The user can delete all Junk files from both sections at once, delete the junk of one section at once, or delete the junk of selected types from both sections. It depends upon user.
Real-time Scanning	This feature works automatically whenever an app or file is installed. The app runs a scan on file to detect viruses, if no virus is found no action is taken. In case virus is detected, the file process stops right there and the user is notified about the risk. This happens within seconds. If the user wants to keep file restriction is removed from it. If the user selects remove option, then file is deleted directly without moving to junk or trash.
Notify for Updates	The system automatically checks for new version of software every 24 hours whenever the user turns on network connection and notifies the user if updates are available.

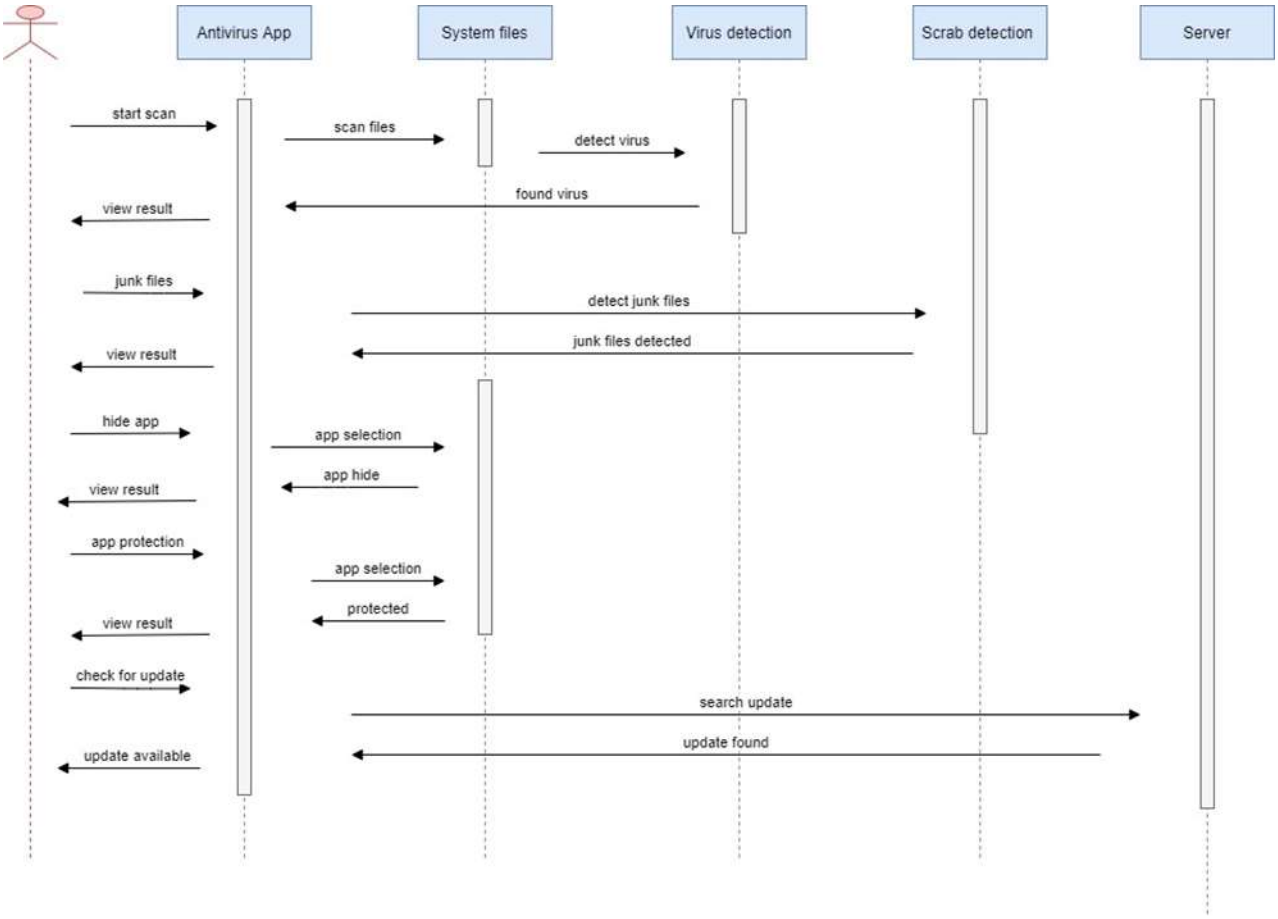
DESIGN & IMPLEMENTATION

DESIGN MODELS

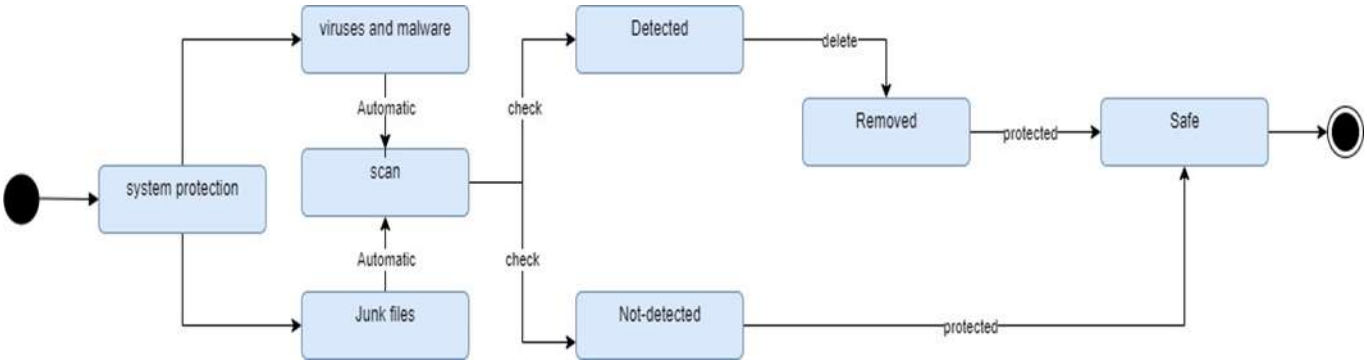
USE CASE DIAGRAM



SEQUENCE DIAGRAM



STATE DIAGRAM

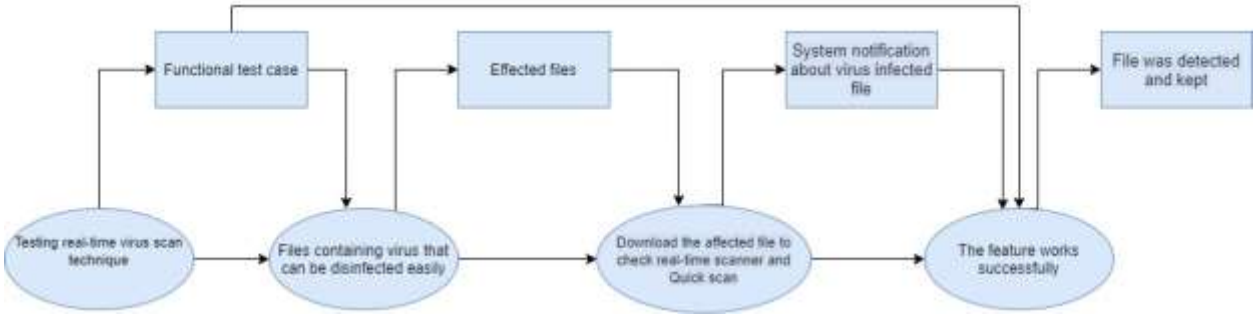


TESTING

TEST CASE 01

REAL-TIME VIRUS-SCANNING TECHNIQUE

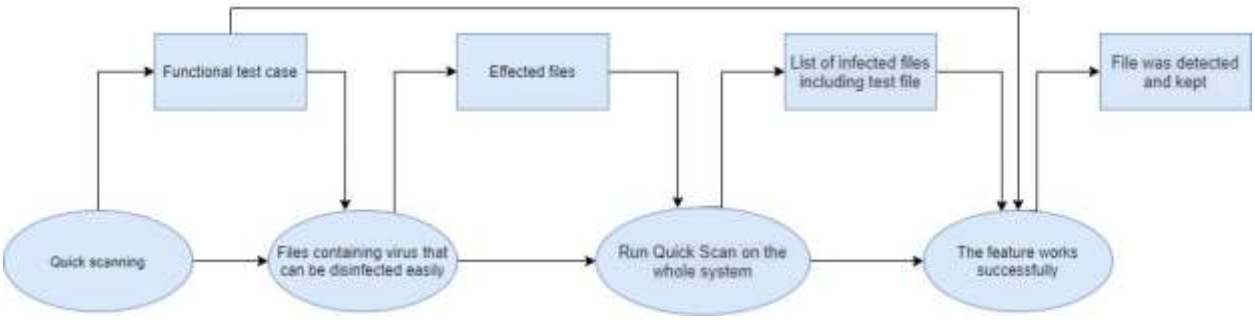
In this test case, we check the real-time scanning feature of our system. For that, we download an infected file into our system. As soon as the download completes, we are notified about the virus affected file. We are given the choice of either deleting or keeping the file. We chose to keep the file for our next test case scenario.



TEST CASE 02

QUICK VIRUS-SCANNING TECHNIQUE

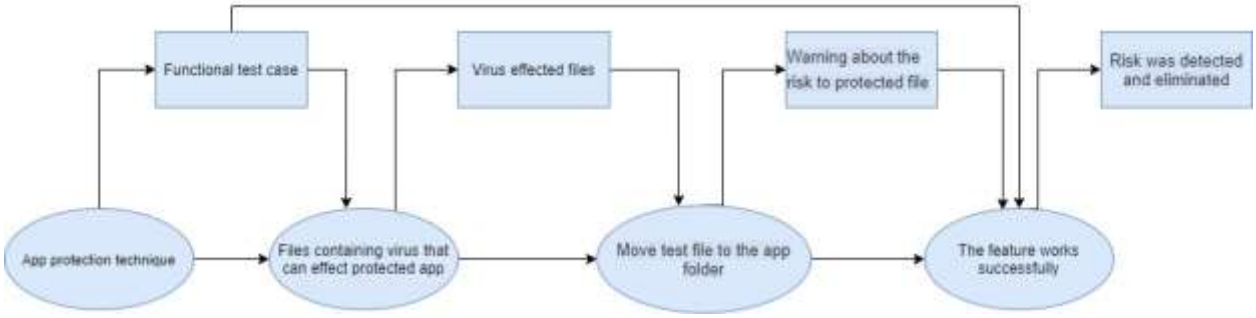
In this test case, we test the quick scan feature of our system. For that, we already have the infected file in our system from our previous test case scenario. So, we run the quick scan and the system gives us the list of the infected files and their location in almost 1 minute. The test file is shown in the list. We are given option to delete or keep infected files. We chose the delete all files option except the test file and confirm this feature works effectively.



TEST CASE 03

APP PROTECTION TECHNIQUE

In this test case, we test the App Protection technique. First, we chose and app and put protection mode on it. To test the feature, we move the file we downloaded in test case scenario 1 and move it into the system folder of the protected app. The system within seconds responds and notifies user about the detection of infected file in protected app and then sends a notification about the deletion of infected file with file name.



TEST CASE 04

JUNK FILES DETECTION TECHNIQUE

In this test case, we test the Junk Files Detection and Deletion technique. For that, we run the Junk Files Detection option in our system. Within a minute, junk files are shown on our screen categorized. We can choose different options to delete junk files. We chose to delete all files.