

SOFTWARE REQUIREMENT SPECIFICATION

**TOPIC: PACS: Permission Abuse Checking System for Android Applications
based on Review Mining**

REVANTH SREEKUMAR

Roll No: 41

1. Introduction

1.1 Purpose

The goal of this work is to discover the Permission Abuse Problems in Android apps. It is achieved by designing a classification scheme that is based on mining the app's descriptions and reviews, classifying the apps into different categories and uses to detect the Permission Abuse Problems.

1.2 Product Scope

According to recent statistics, more than 2.6 million various applications are released in Google Play Store. Unfortunately, due to the limitation of developers' knowledge and the lack of strict development specifications, the quality of the apps cannot be guaranteed. This may lead to potential security problems, especially for the over requirements of the apps' permissions, which is called Permission Abuse Problem. Although some previous studies have already analyzed the permission system, investigated the effectiveness of permission model and attempted to resolve the problem, it still needs an effective and practical concentrated method to detect the permission abuse problem.

2. The Overall Description

2.1.1 Product Perspective

PACS (Permission Abuse Checking System) is based on data and frequent itemsets mining technique to bring an improvement by using the app's reviews and descriptions. PACS firstly classifies the apps into different categories by mining the apps' meta-data, e.g., the reviews, descriptions, etc. Then, it obtains the maximum frequent itemsets and constructs the permission feature database. Finally, PACS is used to detect the unknown application of the abuse of authority.

2.1.2 Operating Environment

Hardware requirements

The minimum hardware requirements are:

- Processor: Intel i3
- Storage : 10 GB Hard Disk space
- Memory : 8 GB RAM

Software requirements

- Operating system: Windows
- Front end: Android, Python
- Back end: MySQL

3. Functional Requirements

Functional requirements represent the indented behaviour of the system. This behaviour may be expressed as services, tasks, or functions that the specified system is required to perform. The following functional requirements have been identified for this project:

- Crawler module: The crawler module collects the app's packages, the descriptions and the reviews from the Google Play Store, and stores them into database.
- Frequent item set mining
- APK decompile

4. Nonfunctional Requirements

4.1 Performance Requirements

The main requirements that the product should be taken into consideration are:

- Accuracy: Accuracy in functioning and the nature of user-friendliness should be maintained in the system.
- Speed: The system must be capable of offering a low computational time.

4.2 Quality Requirements

The most important quality requirements that the system should be taken into consideration are:

- Scalability: The software will meet all of the functional requirements without an unexpected behavior.
- Maintainability: The system should be maintainable. It should keep backups to atone for system failures.
- Reliability: The acceptable threshold for down-time should be long as possible. i.e. mean time between failures should be large as possible. And if the system is broken, time required to get the system back up again should be minimum.

- Testability: The proposed system should be properly tested under various circumstances in order to assure its reliability.

5. Conclusion

This contains full description of the project for the better understanding of what is being intended to do. It lists all the functions performed by the system and the concerns in detail for each of the system functions and actions for the software developer's assistance.