

## A MINI PROJECT REPORT

*for*

**Mini Project in Mobile Application Development (20CSE77A)**

*on*

**Cleaner Plus**

*Submitted by*

# SHIVSWAMI SACHIN CHAVAN

**USN: 1NH20CS285, Sem-Sec: 7-E**

*In partial fulfilment for the award of the degree of*

# BACHELOR OF ENGINEERING

*in*

**COMPUTER SCIENCE AND ENGINEERING**



CERTIFICATE

#### This is to certify that the mini project work titled

**Cleaner Plus**

submitted in partial fulfillment of the degree of Bachelor of Engineering in Computer Science and Engineering by

**SHIVSWAMI SACHIN CHAVAN USN:1NH20CS285**

*DURING*

*ODD SEMESTER 2023-2024*

*for*

*Course: Mini Project in Mobile Application Development-20CSE77A*

#### Signature of Reviewer Signature of HOD

##### SEMESTER END EXAMINATION

*Name of the Examiner Signature with date*

|  |
| --- |
| 1. |
| 2. |

**ABSTRACT**

Cleaner is an Android app that helps your free up space and manage your device. It removes junk files, cache and other unwanted data from your device, so you can get back the storage and performance you need.

It includes a variety of other features to help you keep your phone clean and running smoothly, such as a clipboard cleaner, app manager, whitelist, and customizable cleaning filters. With Cleaner, you can easily keep your phone in top condition, without having to worry about complicated settings or rooting your device.

Our app is designed to be simple and easy to use, while also being fast and lightweight. Plus, it's free and open-source software!

.

# ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be impossible without the mention of the people who made it possible, whose constant guidance and encouragement crowned our efforts with success.

I have great pleasure in expressing gratitude to **Dr. Mohan Manghnani**, Chairman, New Horizon Educational Institutions, for providing necessary infrastructure and creating good environment.

I take this opportunity to express my profound gratitude to **Dr. Manjunatha,** Principal, New Horizon College of Engineering, for his constant support and encouragement.

I would like to thank **Dr. R.J. Anandhi**, Professor and Dean-Academics, NHCE, for his valuable guidance.

I would also like to thank **Dr. B. Rajalakshmi**, Professor and HOD, Department of Computer Science and Engineering, for her constant support.

I also express my gratitude to **Ms. Chempavathy**, Sr. Assistant Professor, Department of Computer Science and Engineering, my mini project reviewer, for constantly monitoring the development of the project and setting up precise deadlines. Her / His valuable suggestions were the motivating factors in completing the work.

##### Shivswami Chavan USN: 1NH20CS285

**CONTENTS**

**ABSTRACT I**

##### [ACKNOWLEDGEMENT II](#_bookmark0)

##### INTRODUCTION

* 1. [PROBLEM DEFINITION **1**](#_bookmark1)
  2. [OBJECTIVES **1**](#_bookmark2)
  3. METHODOLOGY TO BE FOLLOWED **2,3**
  4. HARDWARE AND SOFTWARE REQUIREMENTS **4**

1. **TECHNOLOGY USED**

|  |  |
| --- | --- |
| **2.1** [JAVA TECHNOLOGY](#_bookmark3) | **5,6** |
| **2.2** [XML – EXTENSIBLE MARKUP PAGE](#_bookmark4) | **7** |
| **2.3** KOTLIN | **7** |
|  |  |

1. **DESIGN AND DIAGRAMS**
   1. [DESIGN GOALS **8**](#_bookmark5)
   2. FLOWCHART **9**
2. RESULTS
   1. [HOME SCREEN **10**](#_bookmark15)
   2. CLEAN FUNCTION **10**
   3. ANALYZE FUNCTION **11**
   4. IMAGE OPTIMIZER **11**
   5. WHITELIST **12**
3. CONCLUSION 13

[REFERENCES](#_bookmark32)

1. PLAGIARISM REPORT 14

**CHAPTER 1**

**INTRODUCTION**

* 1. **PROBLEM DEFINITION**

In the current digital age, garbage files and superfluous data can make it difficult for mobile users to manage the storage and performance of their devices. The effectiveness and usability of currently available solutions are frequently out of balance; some are quite complicated, while others compromise user data or bombard consumers with adverts. This emphasizes the need for an open-source mobile application that is safe, easy to use, and effective. One that improves performance, respects user privacy, and makes space management simpler.

* 1. **OBJECTIVES**

This smartphone optimization programmer aims to prioritize a user-friendly interface, optimize device speed, and manage storage effectively by deleting unnecessary files. The implementation of a scheduling function, together with a focus on data security and privacy, is intended to automate regular cleaning operations. In-app lessons teach users how to take care of their devices in the best possible way, and community involvement encourages feedback and teamwork. Examining cross-platform compatibility and frequent updates demonstrate a dedication to customer delight and the special benefits of the software.

## METHODOLOGY TO BE FOLLOWED

#### JAVA:

Java is a popular and flexible programming language for developing mobile apps, providing a stable framework for making apps that function well across a range of platforms. Java is well known for being portable and platform agnostic, allowing programmers to create code once and have it run on several different systems. Through the Android SDK, one of the most widely used mobile operating systems, Android, significantly depends on Java for app development. Because Java is object-oriented, it makes development simpler by allowing for reusable code components and making maintenance simpler.

KOTLIN:

It has become clear that Kotlin is a strong and flexible programming language for creating mobile applications, especially for the Android operating system. Kotlin, acknowledged by Google as the official language for Android, has a clear and expressive syntax that minimizes boilerplate code and increases developer productivity. Developers with current Java codebases may easily convert to it thanks to its compatibility with Java. Modern Kotlin features like extension functions and null safety help to write more dependable and readable code. Because of its smooth interaction with well-known frameworks and development tools, Kotlin is a great option for creating reliable and effective mobile apps. It gives developers the tools they need to efficiently and clearly construct high-quality Android apps.

XML:

The structure and content of user interfaces are defined by the flexible markup language known as XML (eXtensible Markup Language), which is an essential part of the development of mobile apps. XML is frequently used in the context of mobile development to create layouts for Android applications using XML-based layout files. These files make it easier for developers and designers to collaborate by clearly separating the presentation from the business logic. Because of its ease of use and readability, XML is a great option for defining views, expressing user interface elements, and defining attributes like styles and layout parameters. Developers can maintain a modular and maintainable codebase while producing aesthetically pleasing and responsive mobile app interfaces by utilising XML.

## HARDWARE AND SOFTWARE REQUIREMENTS

Hardware System Configuration:

* + - Processor - Intel Core i3/i5/i7
    - RAM - 8GB
    - Hard Disk - 100GB

Software System Configuration:

* + - Operating System: Windows 7/8/10/11
    - Android Studio

## CHAPTER 2

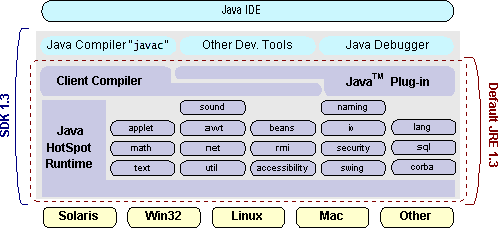
**TECHNOLOGY USED**

* 1. **JAVA TECHNOLOGY**

##### g1To run a programmed on your computer, you typically need to either compile it or interpret it, depending on the programming language. Given that a programmed may be compiled and analyzed, the Java programming language is uncommon. With the compiler, you first transform a programmed into platform-independent Java byte codes, which are then interpreted by the Java platform's interpreter. Each Java byte code instruction is parsed and executed by the interpreter on the computer. Every time the programmed is run, the interpretation occurs rather than a compilation, which only happens once. This is demonstrated in the next illustration.

##### 

##### g3



## XML – Extensible Markup Language

## 

The markup language XML is used to create documents with structured data material (words, images, etc.) and some indication of the function that the material serves are both included in structured information. For instance, the meaning of the content in a section header differs from that of the content in a footnote, which differs from the meaning of the content in a figure caption, or the meaning of the content in a table caption.

* 1. **KOTLIN**

The contemporary programming language Kotlin, created by JetBrains, has quickly become well-known in the software development industry, especially when it comes to creating Android applications. Prominent for its succinct syntax, seamless integration with Java, and improved security attributes, Kotlin streamlines and expedites the development process. Its less boilerplate code, which makes it more expressive and readable than Java, is one of its main advantages. Because Kotlin is completely compatible with Java, developers can use the vast Java ecosystem and easily incorporate it into already-existing projects. Its null safety features improve overall code reliability by lowering the possibility of runtime errors. With Google's backing as the recommended language for Android development, Kotlin has emerged as the top option for creating dependable, effective, and scalable mobile and backend applications.

## CHAPTER 3

**DESIGN AND CASE DIAGRAMS**

## DESIGN GOALS

## User-Friendly Interface: Give top priority to a clear and simple user interface so that users may quickly explore the app without running into any difficulties. Even users with different degrees of technical skill should be able to understand and utilize the design.

## Effective Space Management: Create an application that is very good at locating and eliminating cache, garbage files, and extraneous data. Its main goal is to maximize device storage capacity without sacrificing user data.

## Optimized Performance: Aim to create an application that makes a difference in the speed at which a device loads and responds in general. After running the cleaner app, users should notice a significant increase in the performance and efficiency of their device.

## Privacy and Security: To protect user data while it's being cleaned, put strong security measures in place. Put user privacy first by avoiding invasive data gathering methods and being open and honest about how your data is used.

## Fast and Lightweight: Make an effort to maintain the app's speed and low weight, reducing its influence on device resources and enhancing its capacity for storage optimization and cleaning.

## Clear Communication: Use notifications, alerts, and in-app messages to communicate with users in a simple and succinct manner. Give clear instructions on the operations of the programmed and how they affect the device.

## By aligning the cleaner app with these design goals, the aim is to deliver a reliable, user-friendly, and effective solution for optimizing mobile device performance and storage.

## 

## 

## 3.2 FLOWCHART

START

## 

USER LAUNCHES APP

MAIN MENU

CLEAN OPTION

SCAN COMPLETE?

DISPLAY RESULT

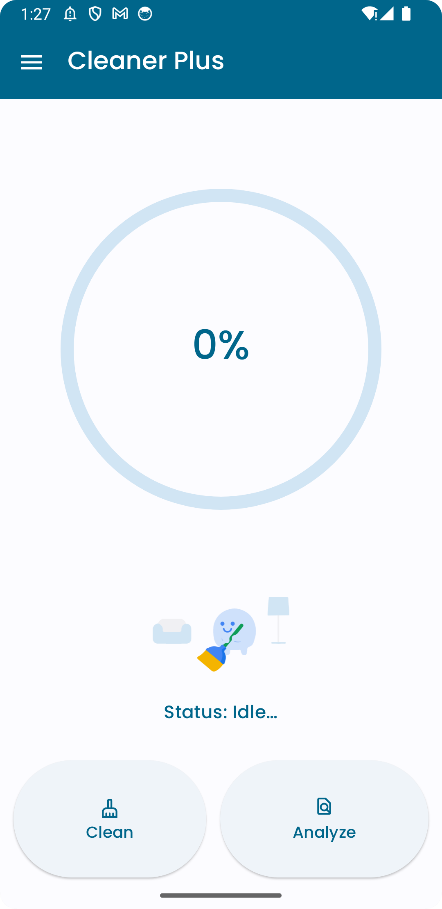
## 

END

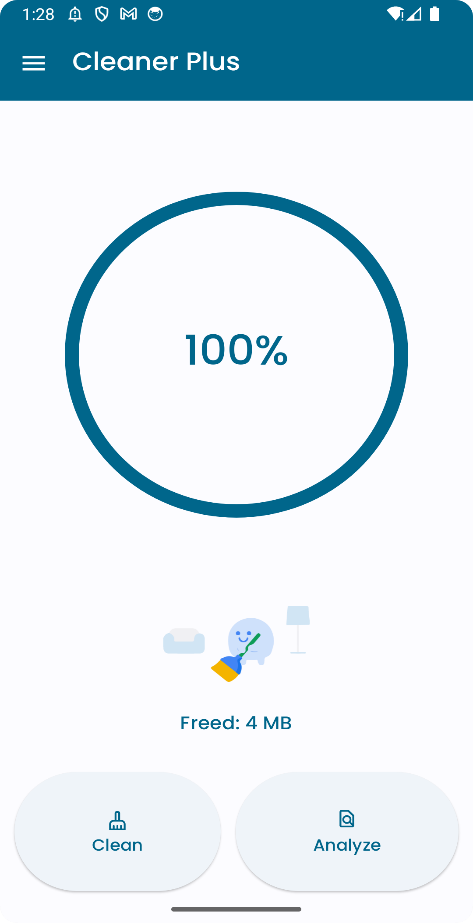
**CHAPTER 4**

**RESULTS**

## HOME SCREEN

****

## 4.2 CLEAN FUNCTION



## ANALYZE FUNCTION

## 

## IMAGE OPTIMIZER

## 

## 

## 4.5 WHITELIST

## 

**CHAPTER**

**CONCLUSION**

The Cleaner app project aims to offer a user-focused and effective way to maximize storage and performance on mobile devices. The application seeks to make the process of clearing space by eliminating superfluous files and data easier to use through an intuitive interface. The design objectives prioritize privacy, security, and a smooth user experience. To help users take care of their devices, features like scheduled cleaning, instructional materials, and community involvement are included. By following these goals, the cleaner app hopes to improve mobile device performance and responsiveness as well as encourage trust and cooperation among its user base. The project's commitment to providing a dependable, adaptable, and efficient solution is further demonstrated by its regular updates and focus to continual progress.

## REFERENCES

## <https://www.w3schools.com/>

## <https://kotlinlang.org/>

## https://github.com/