

Aviral Rajput

A00258751

Assignment Report

Mobile MANAGEMENT SYSTEM

# Abstract

The objective of this project was to develop a Mobile Management System in Java. The purpose of implementing this application was to create a system through which a person can easily add, search customer and vendor and mobiles and generate report accordingly.

Mobile Management System implements The Learning of Binary Search Tree and Hash Table and implementing the learning of design patterns. The application was successfully implemented by using Java programming languages. This application does not aim to target any specific domain but any individual who wants to seek help building small mobile management system, keeping that in mind I tried to keep the interface as simple as possible.

# Introduction

Mobile Management System aims to create a platform where the user can access the system add mobiles and customers and generate the reports. The aim of this project is to create a platform where user can access/interact efficiently with each other and provide ease to keep record efficiently.

# Scope

The scope of the project is limited to the user side. No client can access the platform to create entries and generate reports. Only admin has the right to login and create entries.

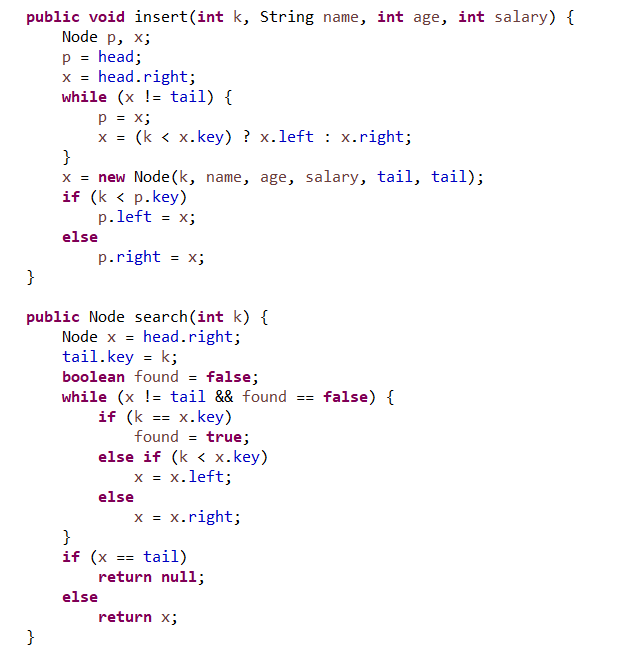
# Limitations and Future Implementation

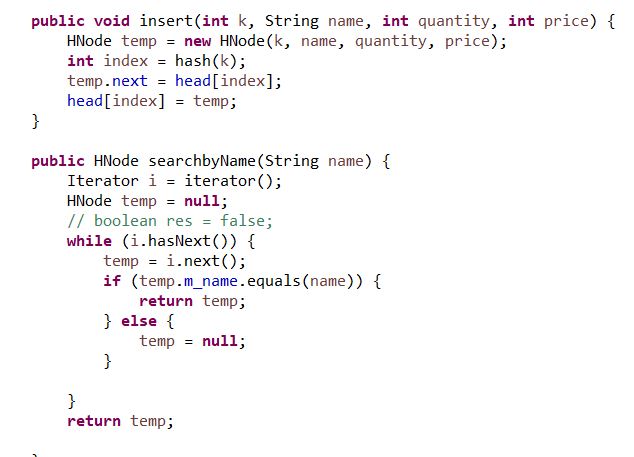
The biggest limitation being on access, the application could be extended to have user interaction and getting out some valuable data from reports.

# Features

* Use of Binary Search Tree and Hash-table data structures to insert and search data.
* Iterator to view data
* Used Singleton Pattern
* Aggregation & Inheritance
* Data passing between each frames on button clicks so that the same list is accessed in all frames.
* Efficient memory management to release resources when required e.g. disposing the frame in the background.

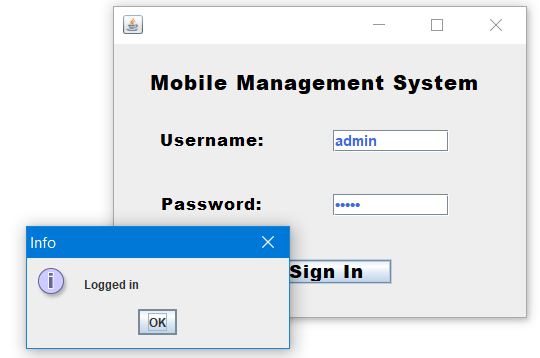
# Code Snippets





# Screenshots

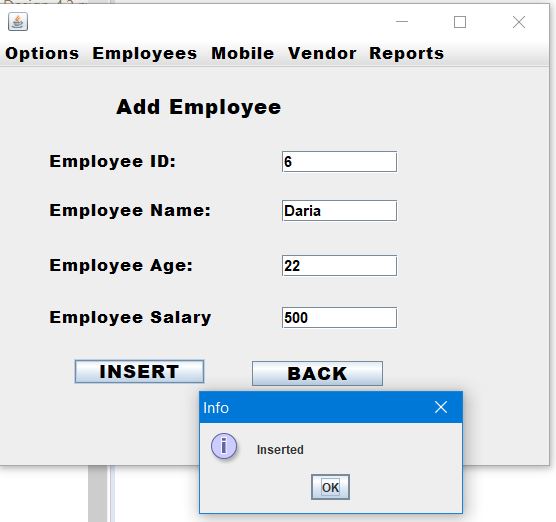
### Login Frame



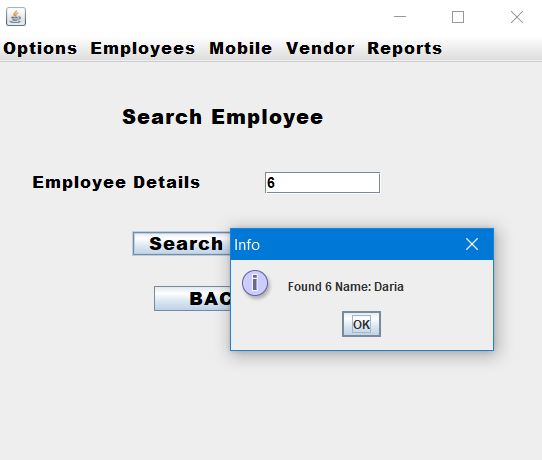
## Main Screen



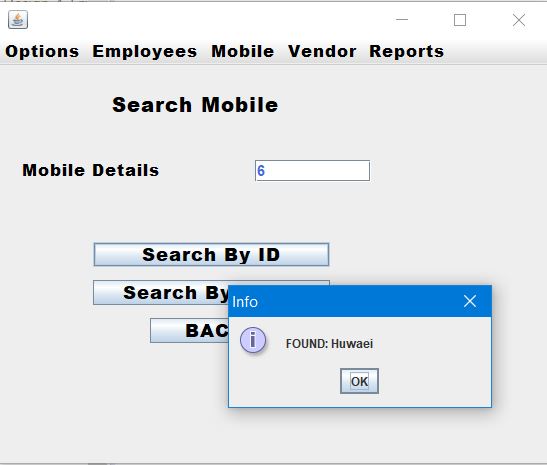
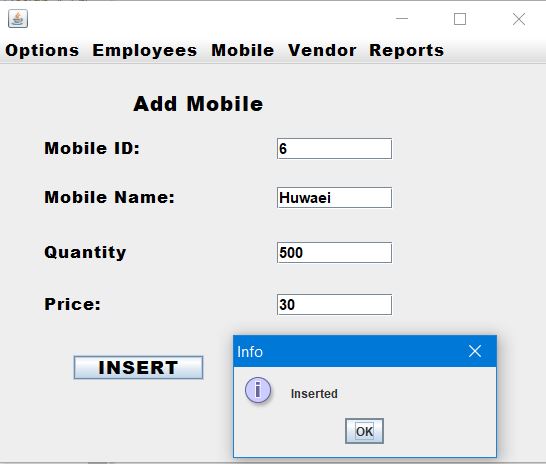
## Add Employee- Uses Binary Search Tree



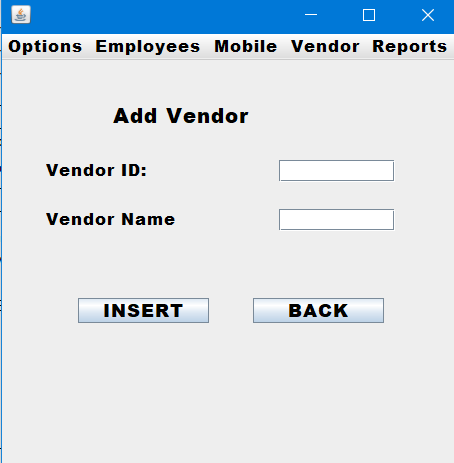
## Search Employee-

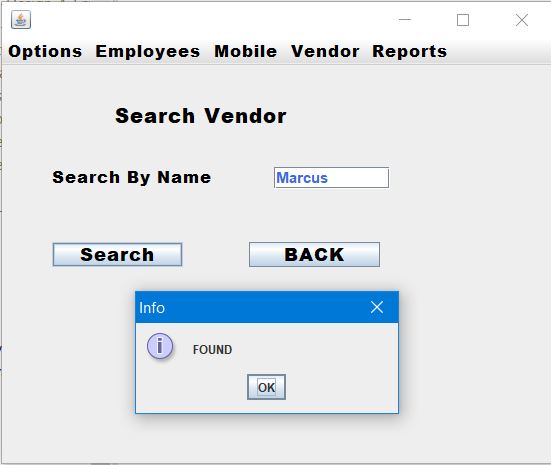


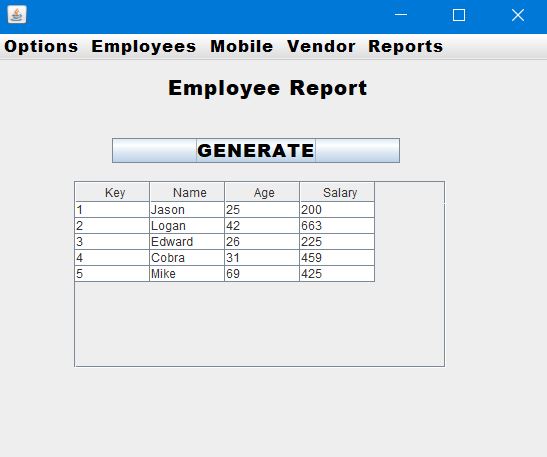
## Add Mobile



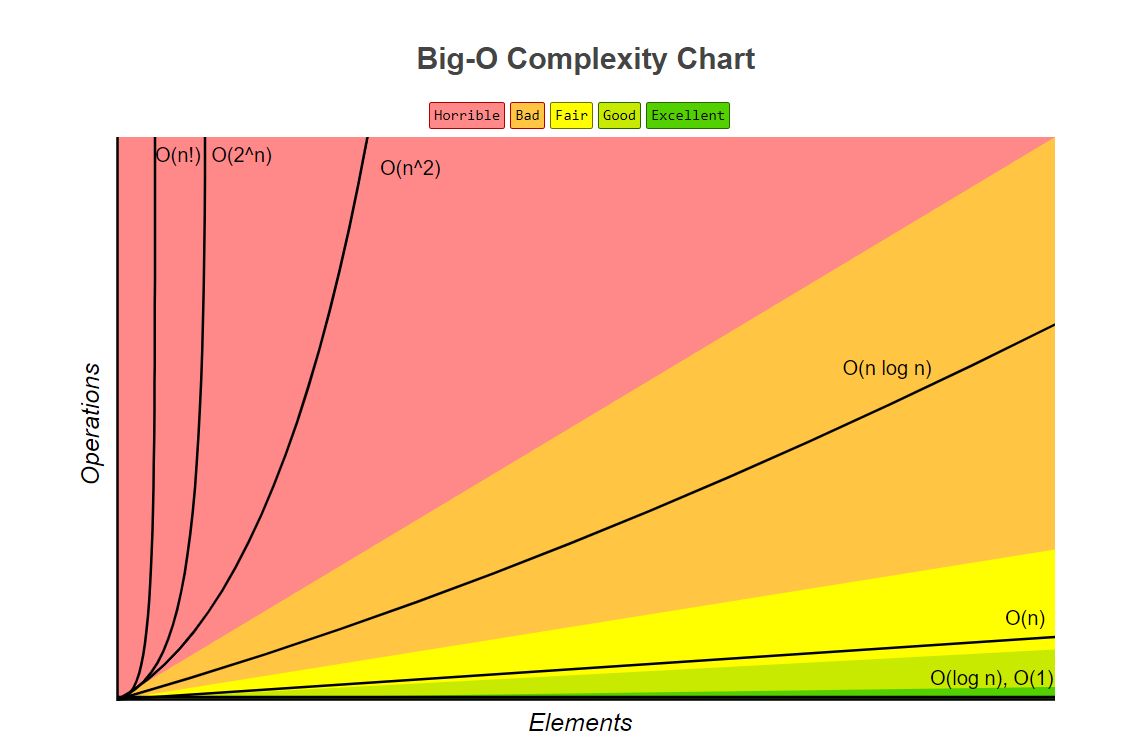
## Add vendor







# Performance



Source: <http://bigocheatsheet.com/>

Cost of search and insert BST.

Time Complexity of BST in **o(log n)** which means it’s a linear function that corresponds to that the time complexity won’t change even if the number of nodes increase.

*Problem*.

Worst-case search and insert are proportional to N. If nodes in order, tree degenerates to linked list.

Cost of search and insert Hash Table.

Theorem.

The Complexity is o(1) is according to the code written.

Problem.

But *O* (1) is achieved only if load-factor is less than 1.  
Worst Case is always (*n*), you can go about looking-up all the elements in the list

# Conclusion

A Simple Mobile management system is created by using Binary Search Tree and Hash table as data structures, values are iterated using Binary Search Tree and different design patterns are applied. Learning outcomes derived from the project helped to discover new features of data structure that will serve for future projects as an avid input.

# References

<https://en.wikibooks.org/wiki/Data_Structures/Hash_Tables>

[BST Complexity](https://www.youtube.com/watch?v=tEoyeoHmqlk)

<https://stackoverflow.com/>

[Comparison of BST and hash table](https://youtu.be/PkO4Fy2R6Oo)

<https://www.geeksforgeeks.org/advantages-of-bst-over-hash-table/>