Dao Vinh Loc

Locdvtcs21019@fpt.edu.vn

Abstract

A program to support teachers manage grade of the students

GrADE MANAGEMENT

Procedural Programming Assignment

# Introduction

This document is to specify technical requirement and solution of Grade Management program to support teachers manage student’s grades.

# Use-case Diagram

Teacher

View grade board

Search

Find Highest grade

Find Lowest grade

Add new grades

Edit existing grades

Remove grades

Sorting grade board

# Technical Architecture

Consoler Screen

Menu Layer

Sub Menu Layer

Core Function Layer

Utility Function Layer

# Function Diagram

Main Menu

1. Grade Board

2. Sorting Grade Board

3. Search

4. Add new grades

5. Edit grades

6. Remove grades

7. Exit Program

1. Sort by Student ID - ASC

2. Sort by Student ID - DESC

3. Sort by Grade – Highest

4. Sort by Grade - Lowest

5. Back to Main Menu

1. Search by Student ID

2. Search by Grade

3. Find Highest Grade

4. Find Lowest Grade

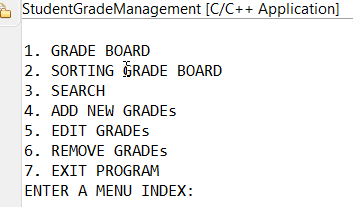
5. Back to Main Menu

# WBS

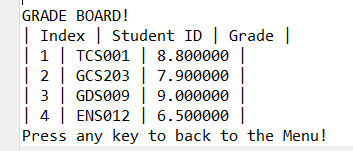
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Layer | Function | Point | Effort |
| 1 | Menu | Main Menu | 1 | 1hrs |
| 2 | Sub Menu | Search | 1 | 1hrs |
| 3 | Sub Menu | Sort | 1 | 1hrs |
| 4 | Core Function | Show (all or one) | 3 | 3hrs |
| 5 | Core Function | Show (array) | 5 | 8hrs |
| 6 | Utility Function | Back | 1 | 1hr |
| 7 | Utility Function | Next | 1 | 1hrs |
| 8 | Core Function | SearchByStudentID | 5 | 4hrs |
| 9 | Core Function | SearchByGrade | 8 | 16hrs |
| 10 | Core Function | Add | 2 | 2hrs |
| 11 | Core Function | Edit | 3 | 4hrs |
| 12 | Core Function | Remove | 2 | 2hrs |
| 13 | Core Function | SortByStudentID | 3 | 4hrs |
| 14 | Core Function | SortByGrade | 3 | 4hrs |
|  |  | **Total** | **39** | **52hrs** |

# Mockup

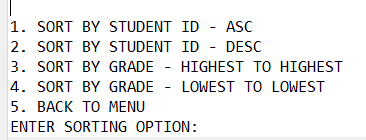
## Main Menu



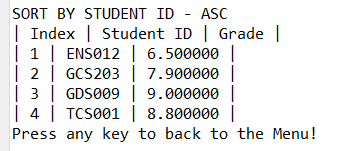
## Grade Board



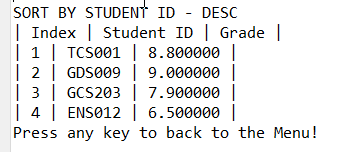
## Sort



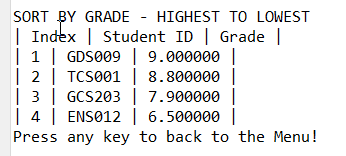
### **SortByStudentID – ASC**



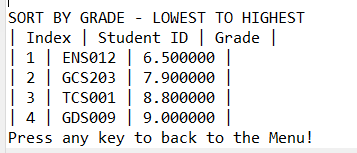
### **SortByStudentID – DESC**



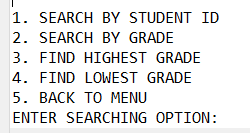
### **SortByGrade – Highest to Lowest**



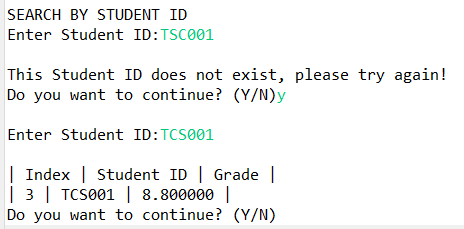
### **SortByGrade – Lowest to Highest**



## Search

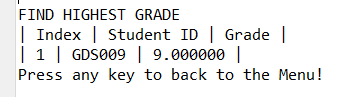


### **SearchByStudentID**

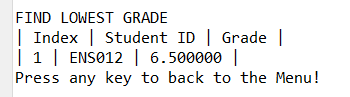


### **SearchByGrade**

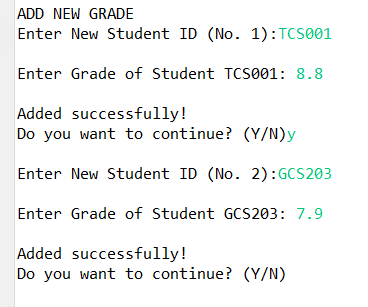
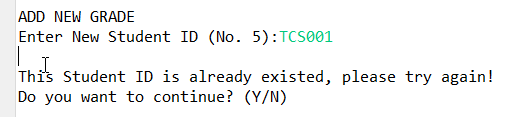
### **FindHighestGrade**



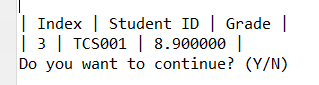
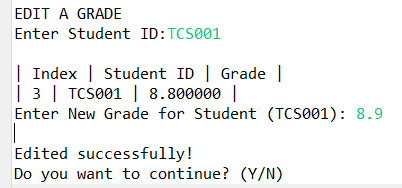
### **FindLowestGrade**



## Add

## Edit



## Remove

