|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Practicum Final Exam – Odd Semester Year 2021/2022** | | | | | | |
| **Subject** | | | **COMP6153001 – Operating System** | | |  |
| **Class** | **:** | **BC08** | | **Start Date** | **: 10 January 2022** |
| **Lecturer** | **:** | **D6353 - Michael Wairisal, S.Kom, M.TI** | | **Start Time** | **: 09:20 WIB** |
| **End Date** | **: 10 January 2022** |
| **End Time** | **: 11:20 WIB** |

**PERATURAN UJIAN:**

*Exam Regulations:*

* Mahasiswa tidak diperbolehkan berdiskusi dan/atau bekerja sama dengan peserta ujian lainnya

*Student is not allowed to discuss and/or work together with other exam participants*

* Mahasiswa tidak diperbolehkan untuk membuka dan menyalin dari **BUKU** atau **CATATAN**, **VIDEO** dari pengajar (recording kelas, VBL, Youtube, dsb) dan **REFERENSI** lainnya

*Student isn't allowed to open and copy from any resources such as notes, videos (class recording, VBL, Youtube, etc) and other references*

* Mahasiswa tidak diperbolehkan membuka dan menyalin jawaban dari internet (google, stackoverflow, dsb)

*Student isn't allowed to open and copy answer from the internet (google, stackoverflow, etc)*

* Asisten **BERHAK** memberi nilai 0 **(NOL)** bagi peserta ujian yang melakukan segala bentuk kecurangan

*Assistant is able to give 0 (ZERO) score for exam participant who does any cheating actions*

* Kumpulkan jawaban tepat pada waktunya, apabila terlambat mengumpulkan maka jawaban tidak akan dikoreksi dan nilai mahasiswa adalah 0

*Submit the answer on time, if not, then the answer will not be checked, and the students will receive 0 (ZERO)*

* Bila Anda tidak membaca peraturan ini, maka Anda dianggap telah membaca dan menyetujuinya

*If you have missed to read these regulations, so you are considered to have read and agreed on it*



**SOFTWARE YANG DIGUNAKAN:**

*Software will be used:*

* Java 8
* Eclipse 2020.6
* NachOS 5.0j

**FILE YANG DIKUMPULKAN:**

*File must be collected:*

* JAVA
* CLASS

**PERHATIAN!**

*Attention!*

* Bagi yang mengerjakan tidak sesuai dengan soal, maka akan diberikan nilai **NOL (0)**

*For those who do not work in accordance with the exam case will be marked as* ***ZERO (0)***

* Bagi yang mengerjakan tidak sesuai dengan software dan versi yang telah ditetapkan, maka akan tetap dikoreksi dengan software dan versi yang telah ditetapkan

*For those who do not work in accordance with the software and specific version will be corrected by the predefined software and version*

* Kompres semua jawaban yang akan diunggah. Pastikan format pengumpulan nama file dan ekstensi sesuai dengan format berikut: **[NIM]-[NAMA].zip**

*Compress all file that will be uploaded. Make sure the format for collecting file name and extension according to the following format:* ***[NIM]-[NAME].zip***

**Important Notes**:

1. Use your **own Console** and **Scheduler (FIFO)**.
2. Youmustuse **Semaphore** to **control** operation.
3. You must use **Timer** to generate tick of time.
4. You must use **File System** to save the data.
5. Use **KThread**.

**Soal**

*Case*

**Shoe Shop**

**Shoe Shop** is a popular local shoe seller that is in Jakarta, Indonesia. You as a programmer are asked to make a program for **Shoe Shop’s admin**. As they are still new in this industry, their first goal is to create a dependable storage system using **nachOS**’ **file system** in **Java programming language**.

In the start of the application, the program will look for a file named **“Shoes.txt”** which will be **converted into list of the items on the store** and load it to our system with the help of file system if the file is available.

After the process has been completed, the application will show 4 menus which represent the features in the application. Which are:

* + **View Shoes**
  + **Add Shoe**
  + **Delete Shoe**
  + **Exit**

Background pattern

Description automatically generated with low confidence

**Figure 1. Menu**

1. **View Shoes (Menu 1)**

The First option of the menu allows the user to **view all data read by file system.** The process follows the following steps which are:

* + - * If **shoe list is empty**, then **show message “No shoes available..”**.



**Figure 2. View validation**

* + - * Otherwise, if the Shoe list **already filled or have list of Shoes,** the application will **show all files received using KThread and Scheduler** (**FIFO Concept**)**.** The item **will be** **shown with delay 1000 milliseconds** (**using Thread sleep**).

Background pattern

Description automatically generated with low confidence

**Figure 3. View all data**

1. **Add Shoe (Menu 2)**

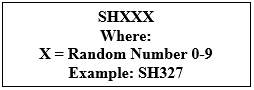
The Second menu option let the user **insert a new Shoe into the system.** This option requires the user to input information regarding the item with the following detail:

* + - * Ask the user to input **Shoe**’**s name**. The Shoe’s name must **end with “shoe”**.
      * Ask the user to input **Shoe**’**s Category**. The Shoe’s category must **between “Sneaker”, “Running”, or “Boot”** (**Case Sensitive**).
      * Ask the user to input **Shoe’s Release Date.** The Shoe’s Release Date which must follow the following format:

**dd-mm-yyyy**

|  |  |  |
| --- | --- | --- |
| Format | Description | Constraint |
| dd | day | Between 1 - 30 |
| mm | month | Between 1 - 12 |
| yyyy | year | Between 2000 - 2022 |

* + - * Ask the user to input **Shoe’s price.** The Shoe’s price must **more than or equals to 5000**.
      * After that, the program will generate **Shoe ID** with the following format:



* + - * After all the required input has been filled, **save the input file named “Shoes.txt”** with the following format.

**[Shoe’s ID]#[Shoe’s Name]#[Shoe’s Category]#[Shoe’s Release Date]#[Shoe’s Price]**

* + - * When the inputted information has been saved, **show success message** to the user.

Text

Description automatically generated

**Figure 4. Input All Data**

**3. Delete Shoe (Menu 3)**

The third menu allows the user to **delete a shoe from the list**. The deleting process follow the following steps:

* + - * If the **shoe list is empty**, then **show message** **“No shoes available”**



**Figure 5. Delete Validation**

* + - * Otherwise, the program will **Print all** the **shoes** in the list and Ask user to **input shoe index**. Validate the index must be **between 1 and total of shoes list**.

A picture containing application

Description automatically generated

**Figure 6. Input Shoe Number (Delete)**

* + - * After that, the program will **delete the shoe** from the list and **show message “Shoe removed!”**



**Figure 7. Success Message**

* + - * After the shoe deletion has been **completed successfully**. **Update the data** inside **“Shoes.txt”** and **redirect the user** to the **main menu.**

**4. Exit (Menu 4)**

* + - * When user choose this menu, the program **will show the time** **in second(s) using a timer**.



**Figure 8. Exit Message**

**Must be collected:**

1. Java Project (Including NachOS and student’s code) compressed (.zip)