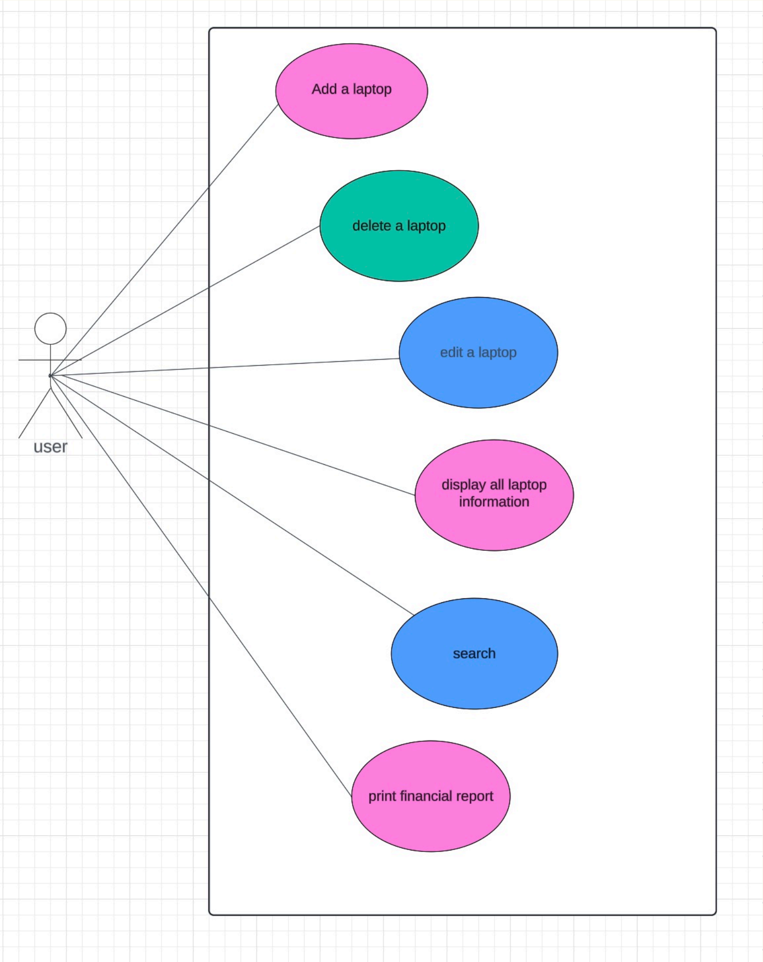
Part 1: Usecase Diagram



(descriptions next page)

1. UC Reference Name: Add a laptop
Overview: Create a new laptop entry in laptop directory with input values of brand, name, repair price, and customer payment
Related use cases: None
Actors: User
Preconditions: User clicks the “AddLaptop” button
Postconditions: A new laptop is added to the laptop repair app

2. UC Reference Name: Delete a laptop
Overview: Delete the repair information of a chosen laptop from the laptop directory
Related use cases: None
Actors: User
Preconditions: There’s at least one laptop in laptop repair directory and “DeleteLaptop” button is clicked
Postconditions: The picked laptop’s parameters are all set to null or 0

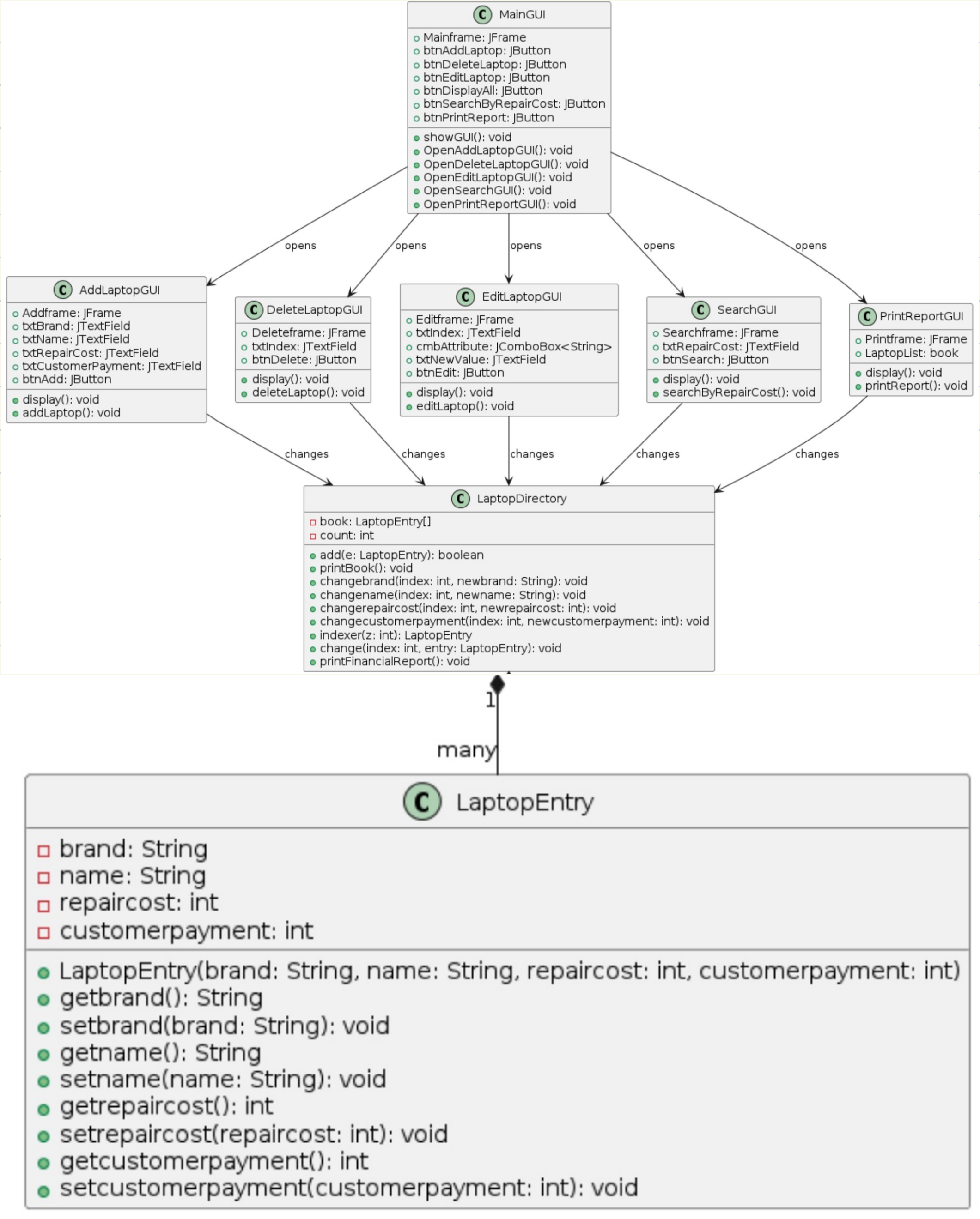
3. UC Reference Name: Edit a laptop
Overview: Modify brand, name, repair price, or customer payment of an existing laptop entry
Related use cases: None
Actors: User
Preconditions: There’s at least one laptop in the laptop repair directory and “EditLaptop” button is clicked
Postconditions: The selected laptop’s details are updated

4. UC Reference Name: Display all laptop information
Overview: Show information of all laptops existing in the laptop repair directory
Related use cases: None
Actors: User
Preconditions: User clicks the “display” button
Postconditions: Print the brand, name, repair price, customer payment of each laptop line by line in order

5. UC Reference Name: Search laptop
Overview: filter searching all laptops whose repair price is above a certain value imputed by the user
Related use cases: None
Actors: User
Preconditions: The compare price is provided and “SearchByRepairCost” button is clicked
Postconditions: Laptops matching the search criteria are displayed

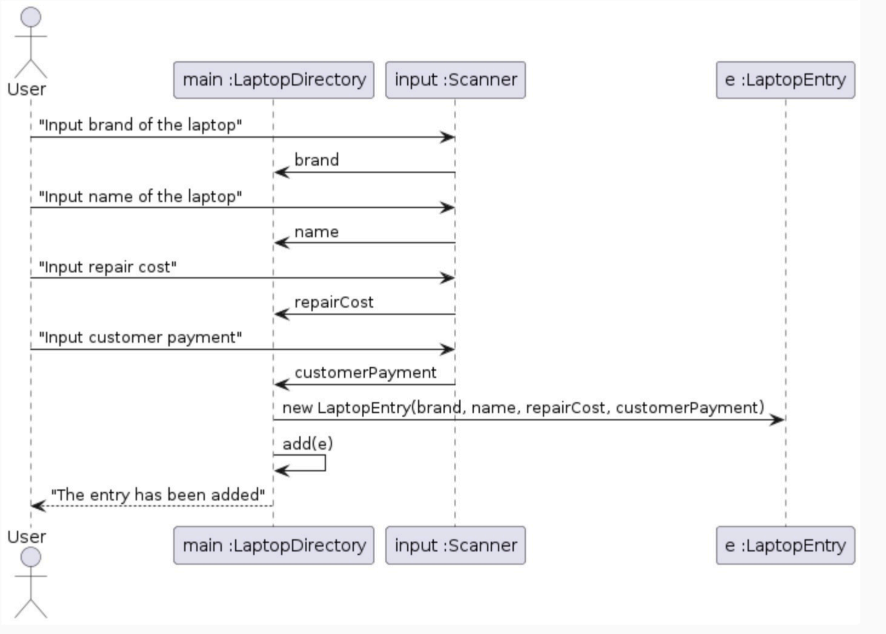
6. UC Reference Name: Print Financial Report
Overview: calculate how much spent, earned, and total profit of the laptop shop for repairing all the laptops
Related use cases: None
Actors: User
Preconditions: User clicks the “PrintReport” button
Postconditions: Generate a financial report that prints the total cost, revenue, and profit for repairing all laptops in the laptop directory

Part 2: Class Diagram

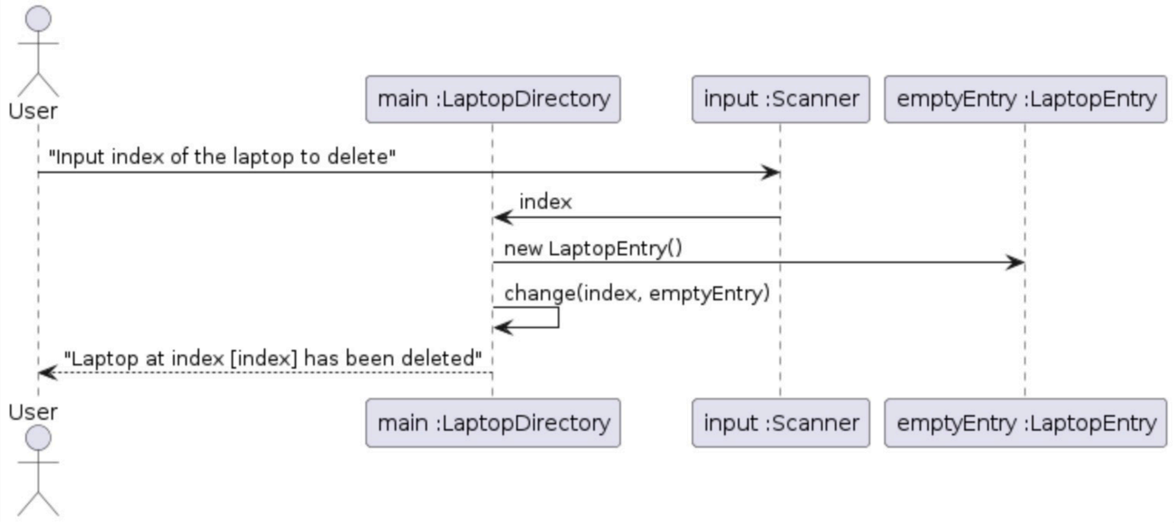

C MainGUI
○ Mainframe: JFrame
○ btnAddLaptop: JButton
○ btnDeleteLaptop: JButton
○ btnEditLaptop: JButton
○ btnDisplayAll: JButton
○ btnSearchByRepairCost: JButton
○ btnPrintReport: JButton
● showGUI(): void
● OpenAddLaptopGUI(): void
● OpenDeleteLaptopGUI(): void
● OpenEditLaptopGUI(): void
● OpenSearchGUI(): void
● OpenPrintReportGUI(): void
opens
opens
opens
opens
opens
C AddLaptopGUI
○ Addframe: JFrame
○ txtBrand: JTextField
○ txtName: JTextField
○ txtRepairCost: JTextField
○ txtCustomerPayment: JTextField
○ btnAdd: JButton
● display(): void
● addLaptop(): void
C DeleteLaptopGUI
○ Deleteframe: JFrame
○ txtIndex: JTextField
○ btnDelete: JButton
● display(): void
● deleteLaptop(): void
C EditLaptopGUI
○ Editframe: JFrame
○ txtIndex: JTextField
○ cmbAttribute: JComboBox<String>
○ txtNewValue: JTextField
○ btnEdit: JButton
● display(): void
● editLaptop(): void
C SearchGUI
○ Searchframe: JFrame
○ txtRepairCost: JTextField
○ btnSearch: JButton
● display(): void
● searchByRepairCost(): void
C PrintReportGUI
○ Printframe: JFrame
○ LaptopList: book
● display(): void
● printReport(): void
changes
changes
changes
changes
changes
C LaptopDirectory
□ book: LaptopEntry[]
□ count: int
● add(e: LaptopEntry): boolean
● printBook(): void
● changebrand(index: int, newbrand: String): void
● changename(index: int, newname: String): void
● changerepaircost(index: int, newrepaircost: int): void
● changecustomerpayment(index: int, newcustomerpayment: int): void
● indexer(z: int): LaptopEntry
● change(index: int, entry: LaptopEntry): void
● printFinancialReport(): void
1
many
C LaptopEntry
□ brand: String
□ name: String
□ repaircost: int
□ customerpayment: int
● LaptopEntry(brand: String, name: String, repaircost: int, customerpayment: int)
● getbrand(): String
● setbrand(brand: String): void
● getname(): String
● setname(name: String): void
● getrepaircost(): int
● setrepaircost(repaircost: int): void
● getcustomerpayment(): int
● setcustomerpayment(customerpayment: int): void

Part 3: Sequence Diagram

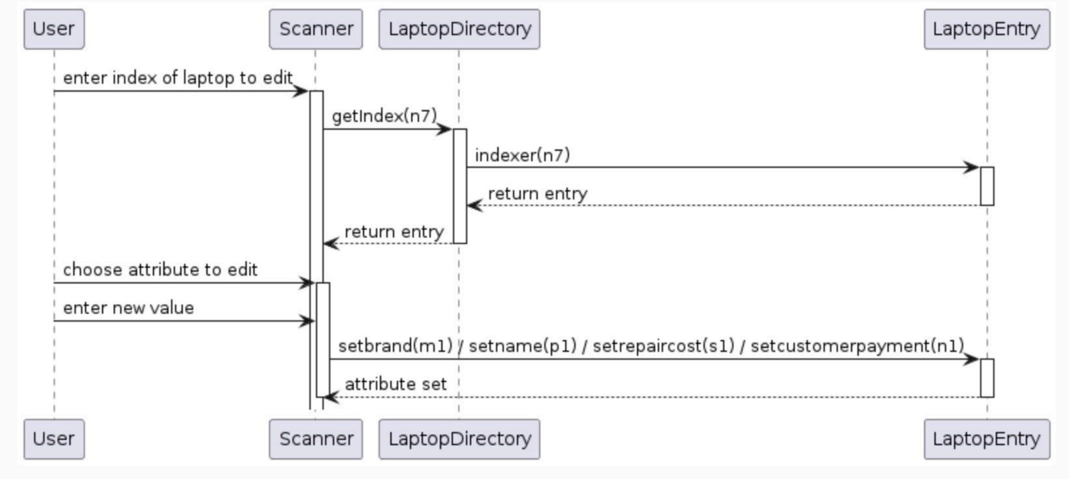
Add a laptop:



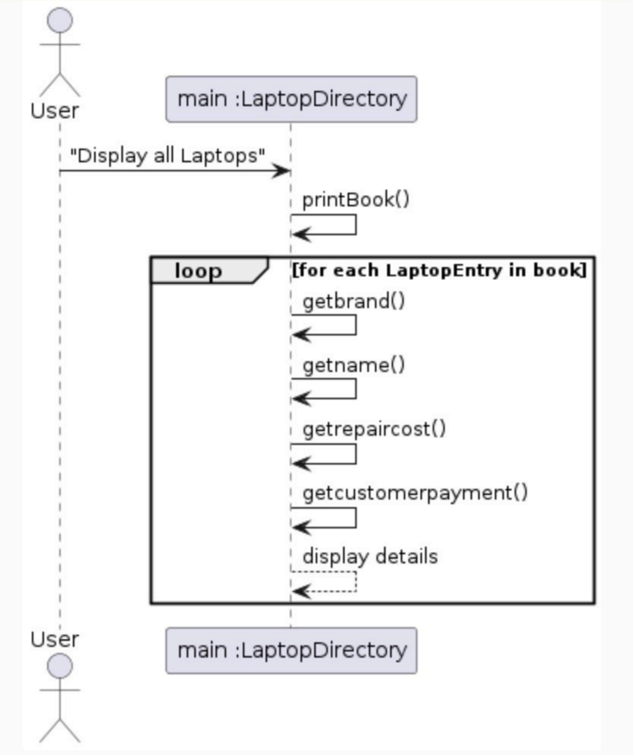
Delete a laptop:



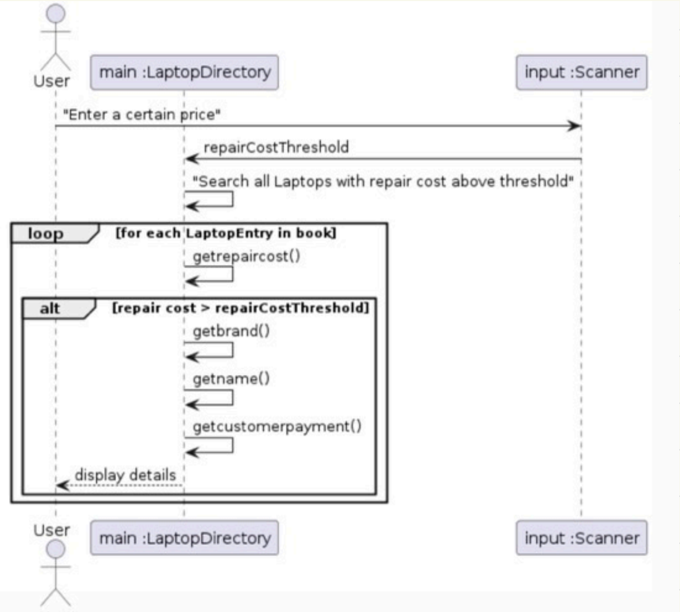
Edit a laptop:



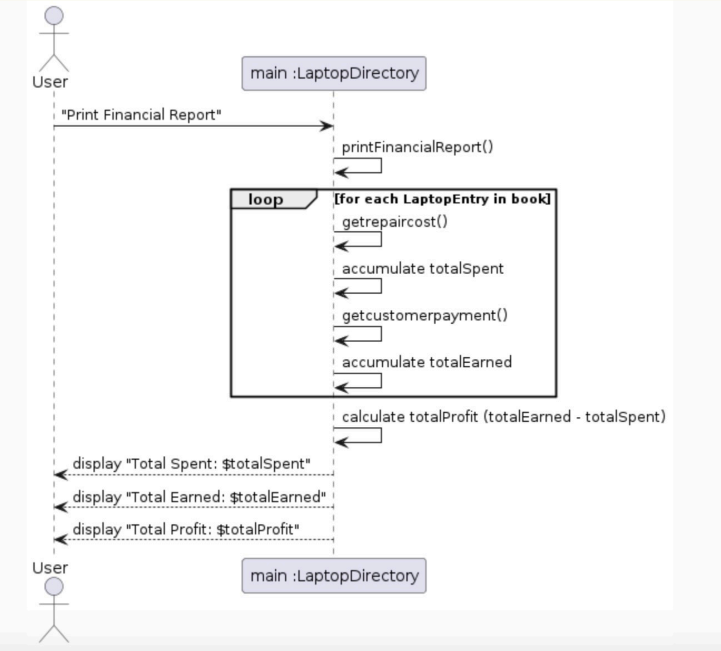
Display all Laptops:



Search laptops above a certain repair price



Print Financial Report


User
main :LaptopDirectory
"Print Financial Report"
printFinancialReport()
loop
[for each LaptopEntry in book]
getrepaircost()
accumulate totalSpent
getcustomerpayment()
accumulate totalEarned
calculate totalProfit (totalEarned - totalSpent)
display "Total Spent: $totalSpent"
display "Total Earned: $totalEarned"
display "Total Profit: $totalProfit"
User
main :LaptopDirectory