BÁO CÁO BÀI THỰC HÀNH SỐ 5

Họ và tên: Đặng Kim Ngân

Mã số sinh viên: 20225751

Mã lớp: 744520

I. New written code

1. **Swing components:** Tạo thư mục GUIProject đặt toàn bộ mã trong package hust.soict.hedspi.swing.

   1.1. **AWTAccumulator:** Tạo AWTAccumulator tạo GUI để nhập vào số và tính tổng các số đã nhập.

```
public class AWTAccumulator extends Frame {
    private TextField tfInput;
    private TextField tfOutput;
    private int sum = 0; // Accumulated sum, init to 0

    // Constructor to setup the GUI components and event handlers;
    public AWTAccumulator() {
        setLayout(new GridLayout(2, 2));

        add(new Label("Enter an Integer: "));

        tfInput = new TextField(10);
        add(tfInput);
        tfInput.addActionListener(new TFInputListener());

        add(new Label("The Accumulated Sum is: "));

        tfOutput = new TextField(10);
        tfOutput.setEditable(false);
        add(tfOutput);

        setTitle("AWT Accumulator");
        setSize(350, 120);
        setVisible(true);
    }

    public static void main(String[] args) {
        new AWTAccumulator();
    }

    private class TFInputListener implements ActionListener {
        @Override
        public void actionPerformed(ActionEvent evt) {
            try {
                int numberIn = Integer.parseInt(tfInput.getText());
                sum += numberIn;
                tfInput.setText("");
                tfOutput.setText(sum + "");
```

1.2. SwingAccumulator: Tạo lớp SwingAccumulator với chức năng tương tự AWTAccumulator

```
public class SwingAccumulator extends JFrame{

    private JTextField tfInput;
    private JTextField tfOutput;
    private int sum=0;          //Accumulated sum, init to 0

    // Constructor to setup the GUI components and event handlers
    public SwingAccumulator() {
        Container cp = getContentPane();
        cp.setLayout(new GridLayout(2,2));

        cp.add(new JLabel("Enter an Integer: "));

        tfInput = new JTextField(10);
        cp.add(tfInput);
        tfInput.addActionListener(new TFInputListener());

        cp.add(new JLabel("The Accumulated Sum is: "));

        tfOutput = new JTextField(10);
        tfOutput.setEditable(false);
        cp.add(tfOutput);

        setTitle("Swing Accumulator");
        setSize(350, 120);
        setVisible(true);
    }
    public static void main(String[] args) {
        new SwingAccumulator();
    }

    private class TFInputListener implements ActionListener {
        @Override
        public void actionPerformed(ActionEvent evt) {
            int numberIn = Integer.parseInt(tfInput.getText());
            sum += numberIn;
            tfInput.setText("");
            tfOutput.setText(sum+"");
```

1.3. So sánh các thành phần Swing và AWT:

- Lập trình với AWT và Swing khá giống nhau (bao gồm các thành phần/container, xử lý sự kiện). Tuy nhiên, có một số khác biệt cần lưu ý:
  • Container cấp cao nhất trong Swing và AWT.
  • Tên lớp của các thành phần trong AWT và tên lớp tương ứng trong Swing.

2. Organizing Swing components with Layout Managers

- Sử dụng Jpanel như secondary-level container để tổ chức lại components.

```
public class NumberGrid extends JFrame {
    private JButton[] btnNumbers = new JButton[10];
    private JButton btnDelete, btnReset;
    private JTextField tfDisplay;

    public NumberGrid() {

        tfDisplay = new JTextField();
        tfDisplay.setComponentOrientation(ComponentOrientation.RIGHT_TO_LEFT);

        JPanel panelButtons = new JPanel(new GridLayout(4, 3));
        addButtons(panelButtons);

        Container cp = getContentPane();
        cp.setLayout(new BorderLayout());
        cp.add(tfDisplay, BorderLayout.NORTH);
        cp.add(panelButtons, BorderLayout.CENTER);

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setTitle("Number Grid");
        setSize(200, 200);
        setVisible(true);
    }

    void addButtons(JPanel panelButtons) {
        ButtonListener btnListener = new ButtonListener();
        for(int i=1; i<=9; i++) {
            btnNumbers[i] = new JButton(""+i);
            panelButtons.add(btnNumbers[i]);
            btnNumbers[i].addActionListener(btnListener);
        }

        btnDelete = new JButton("DEL");
        panelButtons.add(btnDelete);
        btnDelete.addActionListener(btnListener);

        btnNumbers[0] = new JButton("0");
        panelButtons.add(btnNumbers[0]);
```

3. Create a graphical user interface for AIMS with Swing

- Create View Store using Swing
- Create View Cart using JavaFX
- Create Update Store using JavaFX

3.1. Create the StoreScreen class

```java
public class StoreScreen extends JFrame{
    private static Store store = new Store();

    public static void initSetup() {

    public StoreScreen(Store store) {

    public static void main(String[] args) {
        initSetup();
        new StoreScreen(store);
    }

    JPanel createNorth() {
        JPanel north = new JPanel();
        north.setLayout(new BoxLayout(north, BoxLayout.Y_AXIS));
        north.add(createMenuBar());
        north.add(createHeader());
        return north;
    }

    JMenuBar createMenuBar() {
        JMenu menu = new JMenu("Options");

        JMenu smUpdateStore = new JMenu("Update Store");
        smUpdateStore.add(new JMenuItem("Add Book"));
        smUpdateStore.add(new JMenuItem("Add CD"));
        smUpdateStore.add(new JMenuItem("Add DVD"));

        menu.add(smUpdateStore);
        menu.add(new JMenuItem("View store"));
        menu.add(new JMenuItem("View cart"));

        JMenuBar menuBar = new JMenuBar();
        menuBar.setLayout(new FlowLayout(FlowLayout.LEFT));
        menuBar.add(menu);

        return menuBar;
    }
```

3.2. Create MediaStore class

```java
public class MediaStore extends JPanel {
    private Media media;
    public MediaStore(Media media) {

        this.media = media;
        this.setLayout(new BoxLayout(this, BoxLayout.Y_AXIS));

        JLabel title = new JLabel(media.getTitle());
        title.setFont(new Font(title.getFont().getName(), Font.PLAIN, 20));
        title.setAlignmentX(CENTER_ALIGNMENT);

        JLabel cost = new JLabel(""+media.getCost()+"$");
        cost.setAlignmentX(CENTER_ALIGNMENT);

        JPanel container = new JPanel();
        container.setLayout(new FlowLayout(FlowLayout.CENTER));

        // Thêm tương tác cho nút Add to cart
        JButton addToCartButton = new JButton("Add to cart");
        addToCartButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                JOptionPane.showMessageDialog(null, media.getTitle() + "added to cart");
            }
        });
        container.add(addToCartButton);

        // Thêm tương tác cho nút Play
        if(media instanceof Playable) {
            JButton playButton = new JButton("Play");
            playButton.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {

                    JDialog dialog = new JDialog();
                    dialog.setTitle(media.getTitle());
                    dialog.setSize(400, 300);

                    JLabel mediaLabel = new JLabel(media.playGUI());
```

4. JavaFX API

4.1. Create class PainterController

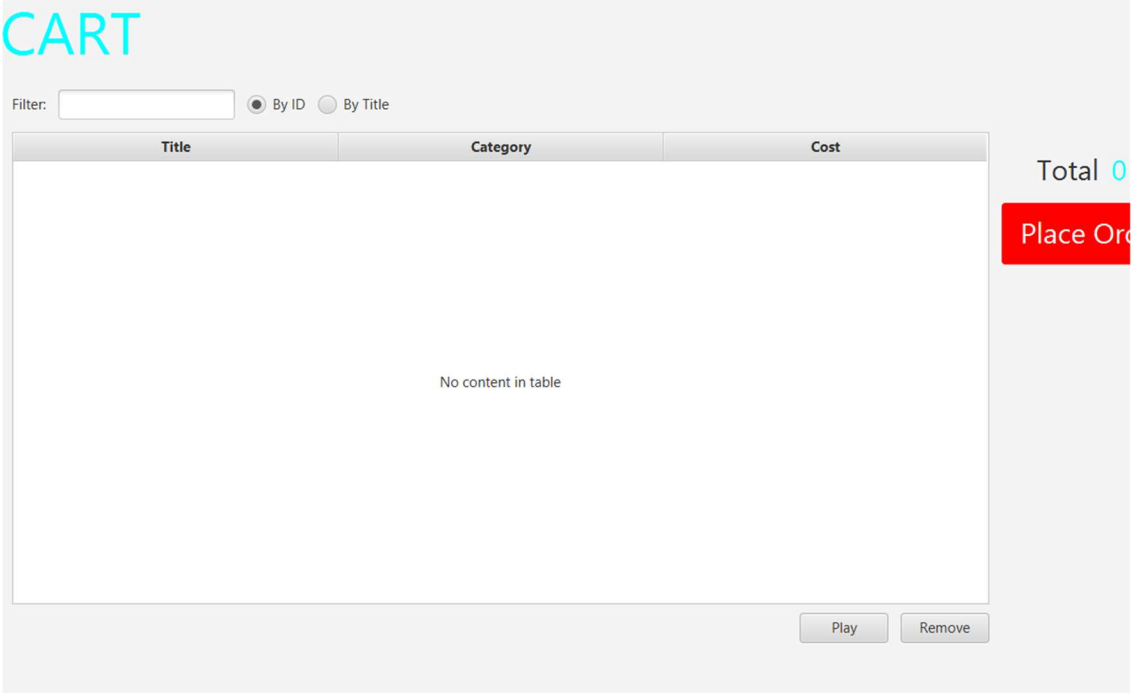
```java
public class PainterController {

    @FXML
    private RadioButton eraser;

    @FXML
    private RadioButton pen;

    @FXML
    private Pane drawingAreaPane;

    // add a ToggleGroup to group RadioButtons
    private ToggleGroup toggleGroup;

    @FXML
    void clearButtonPressed(ActionEvent event) {
        drawingAreaPane.getChildren().clear();
    }

    @FXML
    void drawingAreaMouseDragged(MouseEvent event) {
        Rectangle clipArea = new Rectangle(0, 0, drawingAreaPane.getWidth(), drawingAreaPane.getHeight());
        drawingAreaPane.setClip(clipArea);
        Color inkColor = Color.BLACK;
        if (eraser.isSelected()) {
            inkColor = Color.WHITE;
        }
        Circle newCircle = new Circle(event.getX(), event.getY(), 4, inkColor);
        drawingAreaPane.getChildren().add(newCircle);
    }
```

4.2. Create class Painter

```java
public class Painter extends Application{

    @Override
    public void start(Stage stage) throws Exception {
        Parent root = FXMLLoader.load(getClass()
                .getResource("/hust/soict/hedspi/javafx/Painter.fxml"));

        Scene scene = new Scene(root);
        stage.setTitle("Painter");
        stage.setScene(scene);
        stage.show();
    }
    public static void main(String[] args) {
        launch(args);
    }

}
```

5. Setting up the View Cart Screen with ScreenBuilder

- Tạo cart.fxml trong hust.soict.hedspi.aims.screen.view



6. Integrating JavaFX into Swing application – The JFXPanel class

- Create CartScreen class

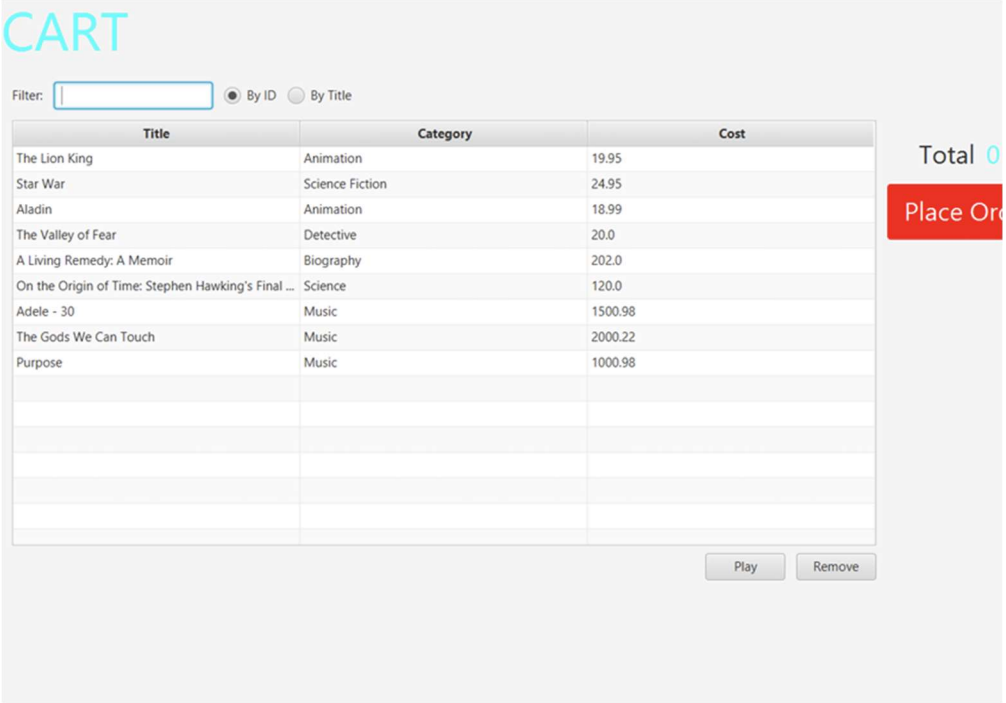
```java
public class CartScreen extends JFrame {
    private static Cart cart = new Cart();

    public CartScreen(Cart cart) {
        super();

        this.cart = cart;

        JFXPanel fxPanel = new JFXPanel();
        this.add(fxPanel);

        this.setTitle("Cart");
        this.setSize(1024, 768);
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        this.setVisible(true);
        Platform.runLater(new Runnable() {
            @Override
            public void run() {
                try {
                    FXMLLoader loader = new FXMLLoader(getClass()
                        .getResource("/huet/soict/hedspi/aims/screen/view/cart.fxml"));

                    CartScreenController controller = new CartScreenController(cart);
                    loader.setController(controller);
                    Parent root = loader.load();
                    fxPanel.setScene(new Scene(root));
                } catch (IOException e) {
                    e.printStackTrace();
                }
            }
        });
    }
```

7. View the items in cart – JavaFX's data-driven UI

- Create CartScreenController class

```java
public class CartScreenController {
    private Cart cart;

    @FXML
    private Button btnPlay;

    @FXML
    private Button btnRemove;

    @FXML
    private TableColumn<Media, Float> colMediaCost;

    @FXML
    private TableColumn<Media, String> colMediaTitle;

    @FXML
    private TableColumn<Media, String> colMediaCategory;

    @FXML
    private TableView<Media> tblMedia;

    @FXML
    private ToggleGroup filterCategory;

    @FXML
    private TextField tfFilter;

    @FXML
    private RadioButton radioBtnFilterId;

    @FXML
    private RadioButton radioBtnFilterTitle;

    @FXML
```

- Kết quả:



8. Updating buttons based on selected item in TableView – ChangeListener

```
@FXML
void placeOrderPressed(ActionEvent event) {
    Alert alert = new Alert(Alert.AlertType.INFORMATION, cart.placeOrder());
    alert.setTitle("Order created");
    alert.setHeaderText(null);
    alert.showAndWait();
}

@FXML
void btnPlayPressed(ActionEvent event) {
    Media media = tblMedia.getSelectionModel().getSelectedItem();
    Alert alert = new Alert(Alert.AlertType.NONE, media.playGUI());
    alert.setTitle("Playing");
    alert.setHeaderText(null);
    alert.getDialogPane().getButtonTypes().add(ButtonType.OK);
    alert.showAndWait();
}
```

9. Deleting a media

```
@FXML
void btnRemovePressed(ActionEvent event) {
    Media media = tblMedia.getSelectionModel().getSelectedItem();
    cart.removeMedia(media);
    costLabel.setText(cart.totalCost() + " $");
}
```

10. Filter items in cart – FilteredList

```
tfFilter.textProperty().addListener(
        new ChangeListener<String>() {

            @Override
            public void changed(ObservableValue<? extends String> observable, String oldValue, String newValue) {
                showFilteredMedia(newValue);
            }

            private void showFilteredMedia(String keyword) {
                FilteredList<Media> filteredList = new FilteredList<>(cart.getItemsOrdered());

                if (!keyword.isEmpty() && radioBtnFilterId.isSelected()) {
                    filteredList.setPredicate(media -> {
                        String idString = String.valueOf(media.getId());
                        return idString.equals(keyword);
                    });
                } else if (!keyword.isEmpty() &&  radioBtnFilterTitle.isSelected()) {
                    filteredList.setPredicate(media -> {
                        String title = media.getTitle().toLowerCase();
                        return title.contains(keyword.toLowerCase());
                    });
                } else {
                    filteredList.setPredicate(null);
                }
                tblMedia.setItems(filteredList);
            }
        });
```

11. Complete the Aims GUI application

11.1. Complete CartScreen

```
@FXML
void placeOrderPressed(ActionEvent event) {
    Alert alert = new Alert(Alert.AlertType.INFORMATION, cart.placeOrder());
    alert.setTitle("Order created");
    alert.setHeaderText(null);
    alert.showAndWait();
}

@FXML
void btnPlayPressed(ActionEvent event) {
    Media media = tblMedia.getSelectionModel().getSelectedItem();
    Alert alert = new Alert(Alert.AlertType.NONE, media.playGUI());
    alert.setTitle("Playing");
    alert.setHeaderText(null);
    alert.getDialogPane().getButtonTypes().add(ButtonType.OK);
    alert.showAndWait();
}
```

- Add method placeOrder in Cart:

```
public String placeOrder() {
    if(itemsOrdered.size()==0) {
            return "Your cart is empty!";
    } else {
            qtyOrdered = 0;
            itemsOrdered.clear();
            return "Order created!\n" + "Now your cart will be empty!";
    }
}
```

- Kết nối StoreScreen với CartScreen

```
public class StoreScreen extends JFrame{
    private static Store store = new Store();
    private static Cart cart = new Cart();
```

```
// Thêm tương tác cho nút Add to cart
JButton addToCartButton = new JButton("Add to cart");
addToCartButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        cart.addMedia(media);
        JOptionPane.showMessageDialog(null, cart.addMedia(media));;
    }
});
container.add(addToCartButton);
```

11.2. Update StoreScreen

11.2.1. Create AddDigitalVideoDiscToStoreScreen class:

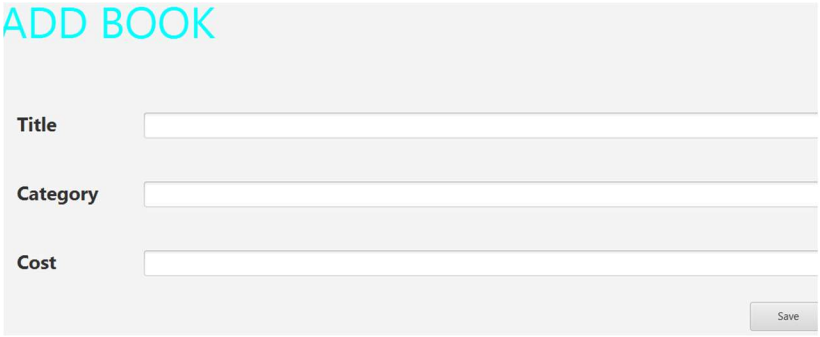
```
public class AddDigitalVideoDiscToStoreScreen extends JFrame{

    private static Store store;

    public static void main(String[] args) {
        new AddDigitalVideoDiscToStoreScreen(store);
    }

    public AddDigitalVideoDiscToStoreScreen(Store store) {

        super();

        AddDigitalVideoDiscToStoreScreen.store = store;

        JFXPanel fxPanel = new JFXPanel();
        this.add(fxPanel);

        this.setTitle("Add DVD");
        this.setSize(1024, 768);
        this.setVisible(true);
        Platform.runLater(new Runnable() {
            @Override
            public void run() {
                try {
                    FXMLLoader loader = new FXMLLoader(getClass().getResource("/hust/soict/hedspi/aims/screen/view/addDVD.fxml"));

                    AddDVDScreenController controller = new AddDVDScreenController(store);
                    loader.setController(controller);
                    Parent root = loader.load();
                    fxPanel.setScene(new Scene(root));
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
```

```java
@FXML
void btnSavePressed(ActionEvent event) {
    String title = tfTitle.getText();
    String category = tfCategory.getText();
    String director = tfDirector.getText();
    int length = 0;
    try {
        length = Integer.parseInt(tfLength.getText());
    } catch (Exception e) {
        Alert alert = new Alert(Alert.AlertType.ERROR, "Failed to parse length!");
        alert.setTitle("Wrong type");
        alert.setHeaderText(null);
        alert.showAndWait();
        return;
    }
    float cost = 0.0f;
    try {
        cost = Float.parseFloat(tfCost.getText());
    } catch (NumberFormatException e) {
        Alert alert = new Alert(Alert.AlertType.ERROR, "Failed to parse cost!");
        alert.setTitle("Wrong type");
        alert.setHeaderText(null);
        alert.showAndWait();
        return;
    }
    DigitalVideoDisc DVD = new DigitalVideoDisc(title, category, director, length,cost);
    store.addMedia(DVD);
    tfTitle.clear();
    tfCategory.clear();
    tfDirector.clear();
    tfLength.clear();
    tfCost.clear();
    Alert alert = new Alert(Alert.AlertType.INFORMATION, "DVD has been added to the strore!");
    alert.setTitle("Success");
    alert.setHeaderText(null);
    alert.showAndWait();
}
```

11.2.2. Create AddBookToStoreScreen class:

```
public class AddBookToStoreScreen extends JFrame{

    private static Store store;

    public static void main(String[] args) {
        new AddBookToStoreScreen(store);
    }

    public AddBookToStoreScreen(Store store) {

        super();

        AddBookToStoreScreen.store = store;

        JFXPanel fxPanel = new JFXPanel();
        this.add(fxPanel);

        this.setTitle("Add Book");
        this.setSize(1024, 768);
        this.setVisible(true);
        Platform.runLater(new Runnable() {
            @Override
            public void run() {
                try {
                    FXMLLoader loader = new FXMLLoader(getClass().getResource("/hust/soict/hedspi/aims/screen/view/addBook.fxml"));

                    AddBookScreenController controller = new AddBookScreenController(store);
                    loader.setController(controller);
                    Parent root = loader.load();
                    fxPanel.setScene(new Scene(root));
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });

    }
```

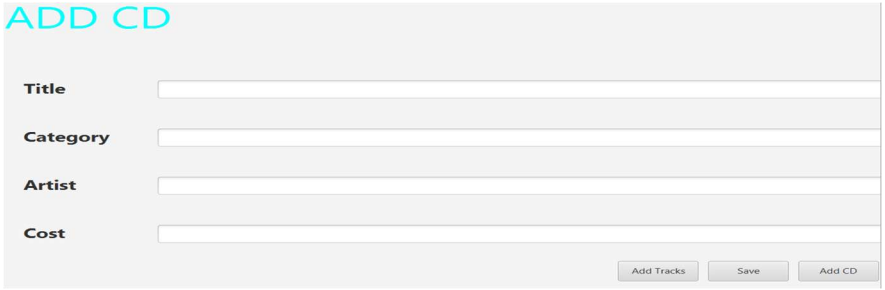
```
public AddBookScreenController(Store store) {
    super();
    this.store = store;
}

@FXML
void btnSavePressed(ActionEvent event) {
    String title = tfTitle.getText();
    String category = tfCategory.getText();
    float cost = 0.0f;
    try {
        cost = Float.parseFloat(tfCost.getText());
    } catch (NumberFormatException e) {
        Alert alert = new Alert(Alert.AlertType.ERROR, "Failed to parse cost!");
        alert.setTitle("Wrong type");
        alert.setHeaderText(null);
        alert.showAndWait();
        return;
    }
    Book book = new Book(title, category, cost);
    store.addMedia(book);
    tfTitle.clear();
    tfCategory.clear();
    tfCost.clear();
    Alert alert = new Alert(Alert.AlertType.INFORMATION, "Book has been added to the strore!");
    alert.setTitle("Success");
    alert.setHeaderText(null);
    alert.showAndWait();
}

@FXML
void initialize() {
    btnSave.setDisable(true);

    tfTitle.textProperty().addListener((observable, oldValue, newValue) -> checkFieldsFilled());
    tfCategory.textProperty().addListener((observable, oldValue, newValue) -> checkFieldsFilled());
    tfCost.textProperty().addListener((observable, oldValue, newValue) -> checkFieldsFilled());
```



11.2.3. Create AddCompactDiscToStoreScreen class

```java
public class AddCompactDiscToStoreScreen extends JFrame{

    private static Store store;

    public static void main(String[] args) {
        new AddCompactDiscToStoreScreen(store);
    }

    public AddCompactDiscToStoreScreen(Store store) {

        super();

        AddCompactDiscToStoreScreen.store = store;

        JFXPanel fxPanel = new JFXPanel();
        this.add(fxPanel);

        this.setTitle("Add CD");
        this.setSize(1024, 768);
        this.setVisible(true);
        Platform.runLater(new Runnable() {
            @Override
            public void run() {
                try {
                    FXMLLoader loader = new FXMLLoader(getClass().getResource("/hust/soict/hedspi/aims/screen/view/addCD.fxml"))

                    AddCDScreenController controller = new AddCDScreenController(store);
                    loader.setController(controller);
                    Parent root = loader.load();
                    fxPanel.setScene(new Scene(root));
                } catch (Exception e) {
                    e.printStackTrace();
                }
```

```java
@FXML
void btnAddCDPressed(ActionEvent event) {
    store.addMedia(CD);
    tfTitle.clear();
    tfCategory.clear();
    tfArtist.clear();
    tfCost.clear();
    btnSave.setDisable(true);
    btnAddCD.setDisable(true);
    btnAddTrack.setDisable(true);
    Alert alert = new Alert(Alert.AlertType.INFORMATION, "CD has been added to the store!");
    alert.setTitle("Success");
    alert.setHeaderText(null);
    alert.showAndWait();
}
@FXML
void btnAddTrackPressed(ActionEvent event) {
    new AddTrack(CD);
}

@FXML
void btnSavePressed(ActionEvent event) {
    String title = tfTitle.getText();
    String category = tfCategory.getText();
    String artist = tfArtist.getText();
    float cost = 0.0f;
    try {
        cost = Float.parseFloat(tfCost.getText());
    } catch (NumberFormatException e) {
        Alert alert = new Alert(Alert.AlertType.ERROR, "Failed to parse cost!");
        alert.setTitle("Wrong type");
        alert.setHeaderText(null);
        alert.showAndWait();
        return;
    }
    CD = new CompactDisc(title, category, artist, cost);
    btnAddCD.setDisable(false);
    btnAddTrack.setDisable(false);
```



11.2.4. Create AddTrackScreen class

```
public class AddTrack extends JFrame{

    private static CompactDisc CD;

    public static void main(String[] args) {
        new AddTrack(CD);
    }

    public AddTrack(CompactDisc CD) {

        super();

        AddTrack.CD = CD;

        JFXPanel fxPanel = new JFXPanel();
        this.add(fxPanel);

        this.setTitle("Add Tracks");
        this.setSize(638, 300);
        this.setVisible(true);
        Platform.runLater(new Runnable() {
            @Override
            public void run() {
                try {
                    FXMLLoader loader = new FXMLLoader(getClass().getResource("/hust/soict/hedspi/aims/screen/view/addTracks.fxml"));

                    AddTrackScreenController controller = new AddTrackScreenController(CD);
                    loader.setController(controller);
                    Parent root = loader.load();
                    fxPanel.setScene(new Scene(root));
                } catch (Exception e) {
                    e.printStackTrace();
```

```
public AddTrackScreenController(CompactDisc CD) {
    super();
    this.CD = CD;
}

@FXML
void btnSaveTrackPressed(ActionEvent event) {
    String title = tfTitle.getText();
    int length = 0;
    try {
        length = Integer.parseInt(tfLength.getText());
    } catch (Exception e) {
        Alert alert = new Alert(Alert.AlertType.ERROR, "Failed to parse length!");
        alert.setTitle("Wrong type");
        alert.setHeaderText(null);
        alert.showAndWait();
        return;
    }
    Track track = new Track(title, length);
    CD.addTrack(track);
    tfTitle.clear();
    tfLength.clear();
    Alert alert = new Alert(Alert.AlertType.INFORMATION, "Track has been added!");
    alert.setTitle("Success");
    alert.setHeaderText(null);
    alert.showAndWait();
}

@FXML
void initialize() {
    btnSaveTrack.setDisable(true);

    tfTitle.textProperty().addListener((observable, oldValue, newValue) -> checkFieldsFilled());
    tfLength.textProperty().addListener((observable, oldValue, newValue) -> checkFieldsFilled());
}
```

12. Check all the previous source codes to catch/handle/delegate runtime exceptions

- Throw Exception cho method addMedia() trong Cart.java và testCart.java

```
public String addMedia(Media media) throws LimitExceededException {
    if (itemsOrdered.size() >= MAX_NUMBERS_ORDERED) {
        throw new LimitExceededException("ERROR: The number of media has reached its limit");
    } else if (itemsOrdered.contains(media)){
        return media.getTitle() + " is already in the cart!";
    } else {
        itemsOrdered.add(media);
        return (media.getTitle() + "has been added!");
    }
}
```

```
public static void main(String[] args) throws LimitExceededException {

    Cart cart = new Cart();

    DigitalVideoDisc dvd1 = new DigitalVideoDisc("The Lion King",
            "Animation", "Roger Allers", 87, 19.95f);
    cart.addMedia(dvd1);

    DigitalVideoDisc dvd2 = new DigitalVideoDisc("Star War",
            "Science Fiction", "George Lucas", 87, 24.95f);
    cart.addMedia(dvd2);
```

- Catch Exception trong MediaStore.java và Aims.java

```
// Thêm tương tác cho nút Add to cart
JButton addToCartButton = new JButton("Add to cart");
addToCartButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        try {
            String message = cart.addMedia(media);
            JOptionPane.showMessageDialog(null, message);
        } catch (LimitExceededException ex) {
            JOptionPane.showMessageDialog(null, ex.getMessage(), "Error", JOptionPane.ERROR_MESSAGE);
        }

    }
});
container.add(addToCartButton);
```

```
switch (option) {
    case 0:
        clearConsole();
        back = true;
        break;
    case 1:
        try {
            cart.addMedia(media);
        } catch (LimitExceededException e) {
            e.printStackTrace();
        }
        break;
```

13. Create a class which inherits from Exception

13.1. Create Player Exception class

```
public class PlayerException extends Exception {
    public PlayerException(String message) {
        super(message);
    }
}
```

13.2. Raise PlayerException in method play()

```
public String playGUI() throws PlayerException {
    return "Playing media";
}
```

```
public String playGUI() throws PlayerException {
    if (this.getLength() > 0) {
            return "Playing DVD: " + this.getTitle() + "\n" +
                "DVD length: " + formatDuration(this.getLength());
        } else {
            throw new PlayerException("ERROR: DVD length is non-positive!");
        }
}
```

```
public String playGUI() throws PlayerException {
    if (this.getLength() > 0) {
        return "Playing track: " + this.getTitle() + "\n" +
            "Track length: " + formatDuration(this.getLength());
    } else {
        throw new PlayerException("ERROR: Track length is non-positive!");
    }
}
```

13.3. Update play() in CompactDisc

```
public String playGUI() throws PlayerException {
    if(this.getLength() > 0) {
        String output =  "Playing CD: " + this.getTitle() + "\n" +
                         "CD length: " + formatDuration(this.getLength()) + "\n"+ "\n";
        for (Track track : tracks) {
            try {
                output += track.playGUI() + "\n";
            } catch (PlayerException e) {
                output += track.getTitle() + "\n" + e.getMessage();
            }
        }
        return output;
        } else {
            throw new PlayerException("ERROR: CD length is non-positive!");
        }
}
```

```java
// Thêm tương tác cho nút Play
if (media instanceof Playable) {
    JButton playButton = new JButton("Play");
    playButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {

            JDialog dialog = new JDialog();
            dialog.setTitle(media.getTitle());
            dialog.setSize(400, 300);

            String mediaInfo = "";
            try {
                mediaInfo = "<html>"+ media.playGUI().replace("\n", "<br/>") + "</html>";
                JLabel mediaLabel = new JLabel(mediaInfo);
                mediaLabel.setVerticalAlignment(JLabel.CENTER);
                mediaLabel.setHorizontalAlignment(JLabel.CENTER);
                JScrollPane scrollPane = new JScrollPane(mediaLabel);
                scrollPane.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLBAR_AS_NEEDED);

                dialog.add(scrollPane);
                dialog.setVisible(true);
            } catch (PlayerException e1) {
                JOptionPane.showMessageDialog(null, e1.getMessage(), "Error", JOptionPane.ERROR_MESSAGE);
            }

        }
    });
    container.add(playButton);
}
```

```java
void btnPlayPressed(ActionEvent event) {
    Media media = tblMedia.getSelectionModel().getSelectedItem();
    Alert alert;
    try {
        alert = new Alert(Alert.AlertType.NONE, media.playGUI());
        alert.setTitle("Playing");
        alert.setHeaderText(null);
        alert.getDialogPane().getButtonTypes().add(ButtonType.OK);
        alert.showAndWait();
    } catch (PlayerException e) {
        alert = new Alert(Alert.AlertType.ERROR, e.getMessage());
        alert.setTitle("ERROR");
        alert.setHeaderText(null);
        alert.showAndWait();
    }

}
```

14. Update Aims class

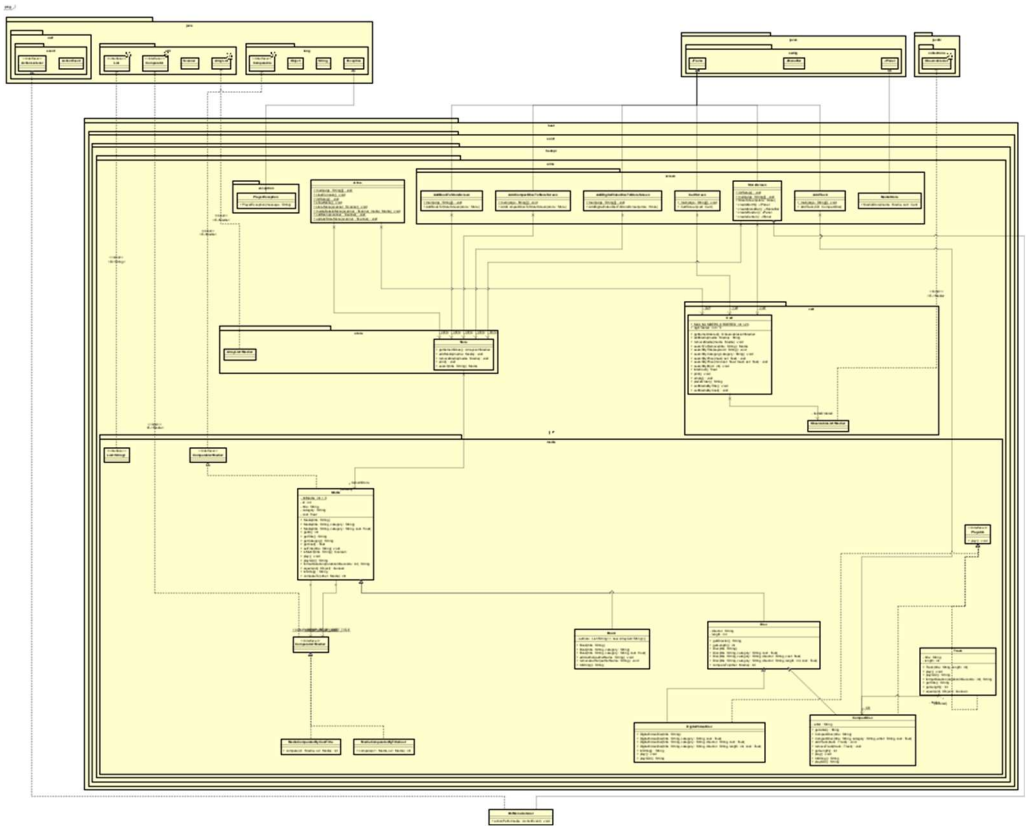
15. Modify the equals() method of Media class

```java
@Override
public boolean equals(Object obj) {
    if (obj == this) {
        return true;
    }
    if (obj == null || !(obj instanceof Media)) {
        return false;
    }
    Media otherMedia = (Media) obj;
    return this.getTitle() != null && this.getTitle().equals(otherMedia.getTitle());

}
```

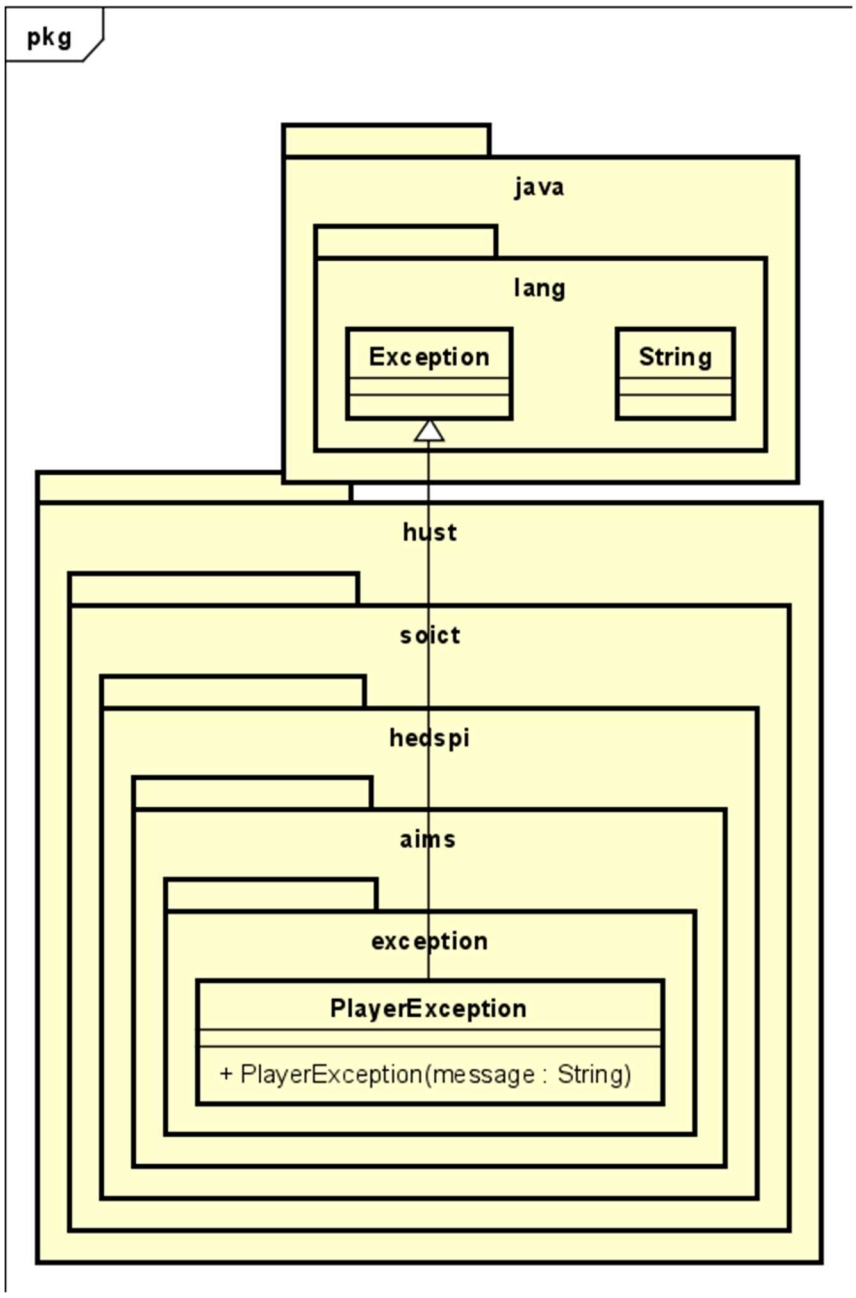
II. UML Diagram

1. Usecase Diagram

2. Class Diagram



3. Exception Hierachical


pkg
java
lang
Exception
String
hust
soict
hedspi
aims
exception
PlayerException
+ PlayerException(message : String)