



CASE STUDY:

LANIYA HOTEL ROOM
RESERVATION SYSTEM

Group 4
1-A NETAD



GROUP MEMBERS

Keievel John,
Mattammu

Tenorio,
Richard

Valdez,
Pierce Harvey

Valerio,
Adam One



SYSTEM DESCRIPTIONS

01

User Age
Verification

02

Destination
Selection

03

Place
Selection

04

Date Selection &
Validation

05

Total no. of Adult

06

Total no. of Child
& Age

SYSTEM DESCRIPTIONS

07

Overall Total no.
of Guest

08

Room Types

09

Room Selection

10

Room Quantity

11

Room Price
Computation

12

Add-On
Selection

SYSTEM DESCRIPTIONS

13

Amenity
Selection

14

Display
Summary

15

Date Format
Validation



OBJECTIVES

Our program aims to create a reservation system that offers different types of rooms, add-ons and amenities that accommodates a certain amount of people



USER TYPES & FEATURES

User Type

User and Guest:

- Requires to provide their information such as name, age, no. of adult & child guest.

The screenshot displays the 'Sparking Hotel Booking System' interface. On the left, the 'User Sign In' screen features a title, 'Username:' and 'Password:' labels, input fields, and 'Back' and 'Next' buttons. On the right, the 'Guest Information' screen is shown, titled 'Enter Guest Details', with 'Number of Adults:' and 'Number of Children:' labels and input fields, and 'Back' and 'Next' buttons. A 'Child Ages' dialog box is open, prompting the user to 'Please enter the age of each child (0-7):' with a 'Child 1 Age:' input field and 'OK' and 'Cancel' buttons. Two red arrows are drawn on the image: one pointing from the 'Next' button of the 'User Sign In' screen to the 'Guest Information' screen, and another pointing from the 'Next' button of the 'Guest Information' screen to the 'Child Ages' dialog box.

Sparking Hotel Booking System

User Sign In

Username:

Password:

Sparking Hotel Booking System

Guest Information

Enter Guest Details

Number of Adults:

Number of Children:

Child Ages

Please enter the age of each child (0-7):

Child 1 Age:

USER TYPES & FEATURES

User Type

Guest or Current User

- Ask user to select destination,
- Check-In Check-Out date..

The image displays three overlapping screenshots of the 'Sparking Hotel Booking System' interface, illustrating the user flow. Red arrows indicate the progression from the first screen to the second, and then to the third.

Destination Selection (Left Screenshot): This screen features two buttons: 'Local' and 'International'. At the bottom, there are 'Back' and 'Next' navigation buttons.

Destination Selection (Middle Screenshot): This screen shows the 'Local' and 'International' buttons. Below them, there are four radio button options for destination selection: 'Hong Kong', 'Japan', 'Singapore', and 'South Korea'. 'Back' and 'Next' buttons are at the bottom.

Destination Selection (Right Screenshot): This screen shows the 'Local' and 'International' buttons. Below them, there are four radio button options: 'Baguio', 'Boracay', 'El Nido', and 'Siargao'. 'Back' and 'Next' buttons are at the bottom.

Check-In and Check-Out Information (Bottom Screenshot): This screen is titled 'Check-In and Check-Out Information'. It contains a 'Booking Details' section with two input fields: 'Check-in Date (MM-DD-YYYY):' and 'Check-out Date (MM-DD-YYYY):'. 'Back' and 'Next' buttons are at the bottom.

USER TYPES & FEATURES

User Type

Guest or Current User

- Ask user to select room and special offers. (add-on/amenities).

The screenshot displays the 'Sparking Hotel Booking System' interface. On the left, the 'Room Suggestions' section lists five room types: Standard, Deluxe, Quadruple, Family, and Suite. The 'Family' room type is selected, showing a capacity of 6 guests with 1 extra bed. A red arrow points from this selection to the 'Select Amenities' section. The 'Select Amenities' section lists six amenities with their respective prices and selection counts: Thai Massage (P1540), Aroma Facial Massage (P1045), Access to Pool (P300), Gym Access (P500), and Foot Spa Services (P825). A red arrow points from the 'Gym Access' amenity to the 'Select Add-ons' section. The 'Select Add-ons' section lists five add-ons with their respective prices and selection counts: Bed (P650.00/night), Blanket (P250.00/night), Pillow (P100.00/night), and Toiletries (P200.00/set). At the bottom of each section are 'Back' and 'Next' buttons.

Room Suggestions

Suggested Room type is Double

Standard

☐ Capacity: 1 guest with 1 extra bed

Deluxe

☐ Capacity: 2 guests with 1 extra bed

Quadruple

☐ Capacity: 4 guests with 1 extra bed

Family

☒ Capacity: 6 guests with 1 extra bed

Suite

☐ Capacity: 4 guests

Select Amenities

Available Amenities

Thai Massage (P1540):	<input type="text" value="0"/>	P0.00
Aroma Facial Massage (P1045):	<input type="text" value="1"/>	P1045.00
Access to Pool (P300):	<input type="text" value="1"/>	P300.00
Gym Access (P500):	<input type="text" value="1"/>	P500.00
Foot Spa Services (P825):	<input type="text" value="0"/>	P0.00

Select Add-ons

Available Add-ons

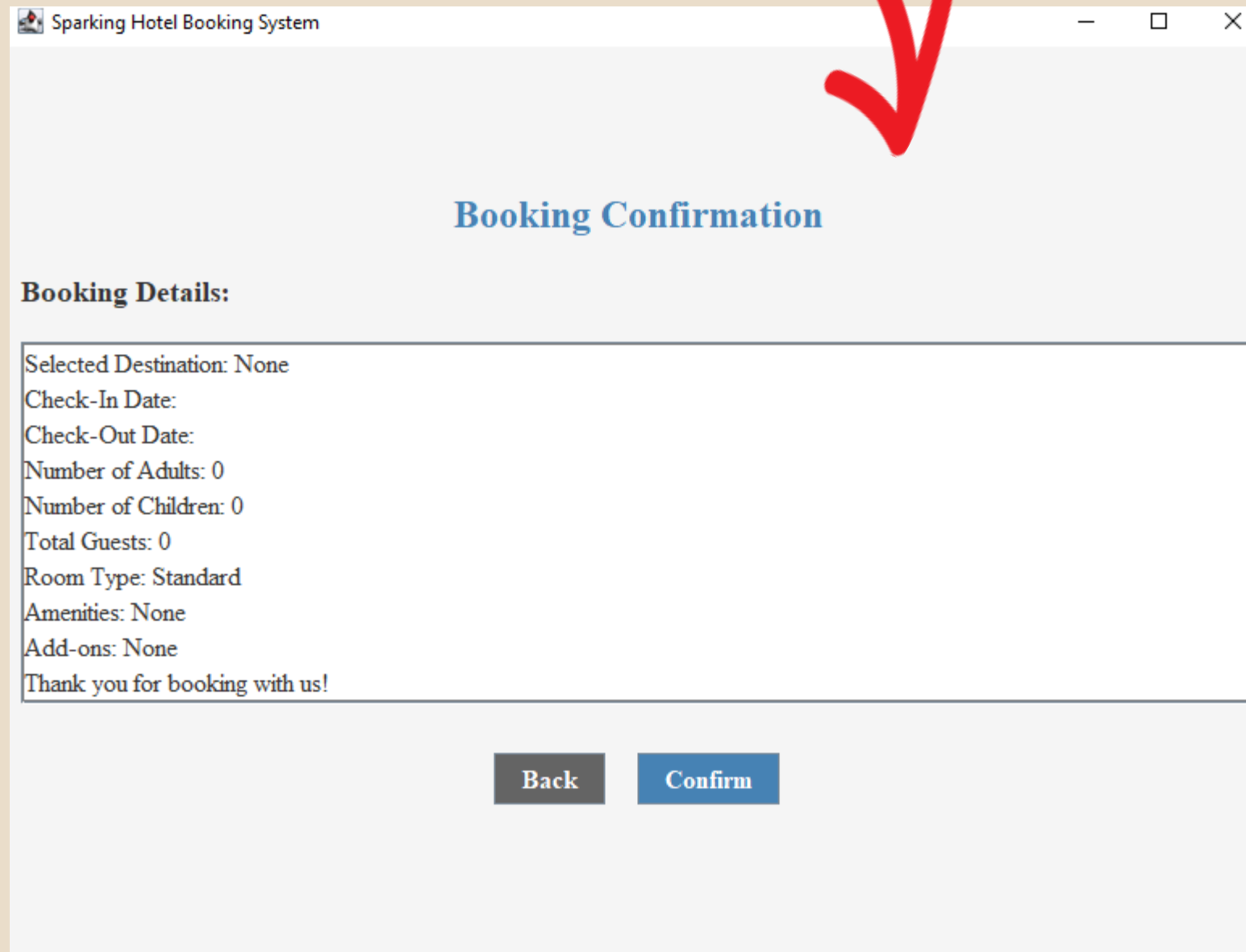

Bed (P650.00/night):	<input type="text" value="0"/>	P0.00
Blanket (P250.00/night):	<input type="text" value="1"/>	P250.00
Pillow (P100.00/night):	<input type="text" value="1"/>	P100.00
Toiletries (P200.00/set):	<input type="text" value="0"/>	P0.00

Back **Next**

Back **Next**

USER TYPES & FEATURES

Confirmation Panel:



Sparking Hotel Booking System

Booking Confirmation

Booking Details:


Selected Destination: None
Check-In Date:
Check-Out Date:
Number of Adults: 0
Number of Children: 0
Total Guests: 0
Room Type: Standard
Amenities: None
Add-ons: None
Thank you for booking with us!

[Back](#) [Confirm](#)

USER TYPES & FEATURES

Features

- Date Validation
- Determines the Season



```
// Validate dates
try {
    SimpleDateFormat dateFormat = new SimpleDateFormat("MM-dd-yyyy");
    dateFormat.setLenient(false);

    Date currentDate = new Date();
    Date checkInDate = dateFormat.parse(checkIn);
    Date checkOutDate = dateFormat.parse(checkOut);

    if (checkInDate.before(currentDate)) {
        JOptionPane.showMessageDialog(panel, "Check-in date cannot be in the past.", "Validation Error", JOptionPane.ERROR_MESSAGE);
        return;
    }

    if (!checkOutDate.after(checkInDate)) {
        JOptionPane.showMessageDialog(panel, "Check-out date must be after the check-in date.", "Validation Error", JOptionPane.ERROR_MESSAGE);
        return;
    }
} catch (ParseException ex) {
    JOptionPane.showMessageDialog(panel, "Please input valid dates in the format MM-DD-YYYY.", "Validation Error", JOptionPane.ERROR_MESSAGE);
    return;
}

// Validation passed
checkInDate = checkIn;
checkOutDate = checkOut;

// Navigate to the next panel
cardLayout.show(cardPanel, "5"); // Replace "2" with the destination panel key
});

buttonPanel.add(backButton);
buttonPanel.add(nextButton);

panel.add(buttonPanel, BorderLayout.SOUTH);

return panel;
}
```

USER TYPES & FEATURES

Features

- Room suggestion
- Ability to skip add-on/amenities

```
JRadioButton radioButton = new JRadioButton(room[1]);
radioButton.setFont(new Font("Serif", Font.PLAIN, 14)); // Adjusted font size for radio button text
radioButton.setBackground(new Color(245, 245, 245));
roomGroup.add(radioButton);

roomPanel.add(radioButton, BorderLayout.WEST);
panel.add(roomPanel);
panel.add(Box.createRigidArea(new Dimension(0, 10))); // Spacer between room panels
}

panel.add(Box.createRigidArea(new Dimension(0, 20))); // Spacer before buttons

// Navigation Buttons
JPanel buttonPanel = new JPanel(new FlowLayout(FlowLayout.CENTER, 20, 10));
buttonPanel.setBackground(new Color(245, 245, 245));

JButton backButton = new JButton("Back");
JButton nextButton = new JButton("Next");

backButton.setFont(new Font("Serif", Font.BOLD, 14)); // Adjusted button font
nextButton.setFont(new Font("Serif", Font.BOLD, 14)); // Adjusted button font
backButton.setBackground(new Color(100, 100, 100));
nextButton.setBackground(new Color(70, 130, 180));
backButton.setForeground(Color.WHITE);
nextButton.setForeground(Color.WHITE);

// Action Listeners for navigation
backButton.addActionListener(e -> cardLayout.show(cardPanel, "5")); // Back to Destination Panel
nextButton.addActionListener(
    e -> {
        // Validation to ensure a room is selected
        if (roomGroup.getSelection() == null) {
            JOptionPane.showMessageDialog(panel, "Please select a room type before proceeding.", "Validation Error", JOptionPane.ERROR_MESSAGE);
            return;
        }

        // If validation passes, navigate to the next panel
        cardLayout.show(cardPanel, "7"); // Go to Amenities Panel
    });

buttonPanel.add(backButton);
buttonPanel.add(nextButton);
panel.add(buttonPanel, BorderLayout.SOUTH);

return panel;
}
```

```
private JPanel createRoomSuggestionPanel() {
    JPanel panel = new JPanel();
    panel.setLayout(new BoxLayout(panel, BoxLayout.Y_AXIS));
    panel.setBackground(new Color(245, 245, 245)); // Light gray background
    panel.setBorder(BorderFactory.createEmptyBorder(20, 20, 20, 20)); // Padding around the panel

    // Title
    JLabel title = new JLabel("Room Suggestions", SwingConstants.CENTER);
    title.setFont(new Font("Serif", Font.BOLD, 20)); // Adjusted font size
    title.setAlignmentX(Component.CENTER_ALIGNMENT);
    title.setBorder(BorderFactory.createEmptyBorder(10, 0, 20, 0));
    panel.add(title);

    panel.add(Box.createRigidArea(new Dimension(0, 20))); // Spacer

    //text
    JLabel text = new JLabel("Suggested Room type is Double", SwingConstants.LEFT);
    title.setFont(new Font("Serif", Font.BOLD, 20)); // Adjusted font size
    title.setAlignmentX(Component.LEFT_ALIGNMENT);
    title.setBorder(BorderFactory.createEmptyBorder(10, 0, 20, 0));
    panel.add(text);

    panel.add(Box.createRigidArea(new Dimension(0, 20))); // Spacer

    // Button Group for Room Selection
    ButtonGroup roomGroup = new ButtonGroup();

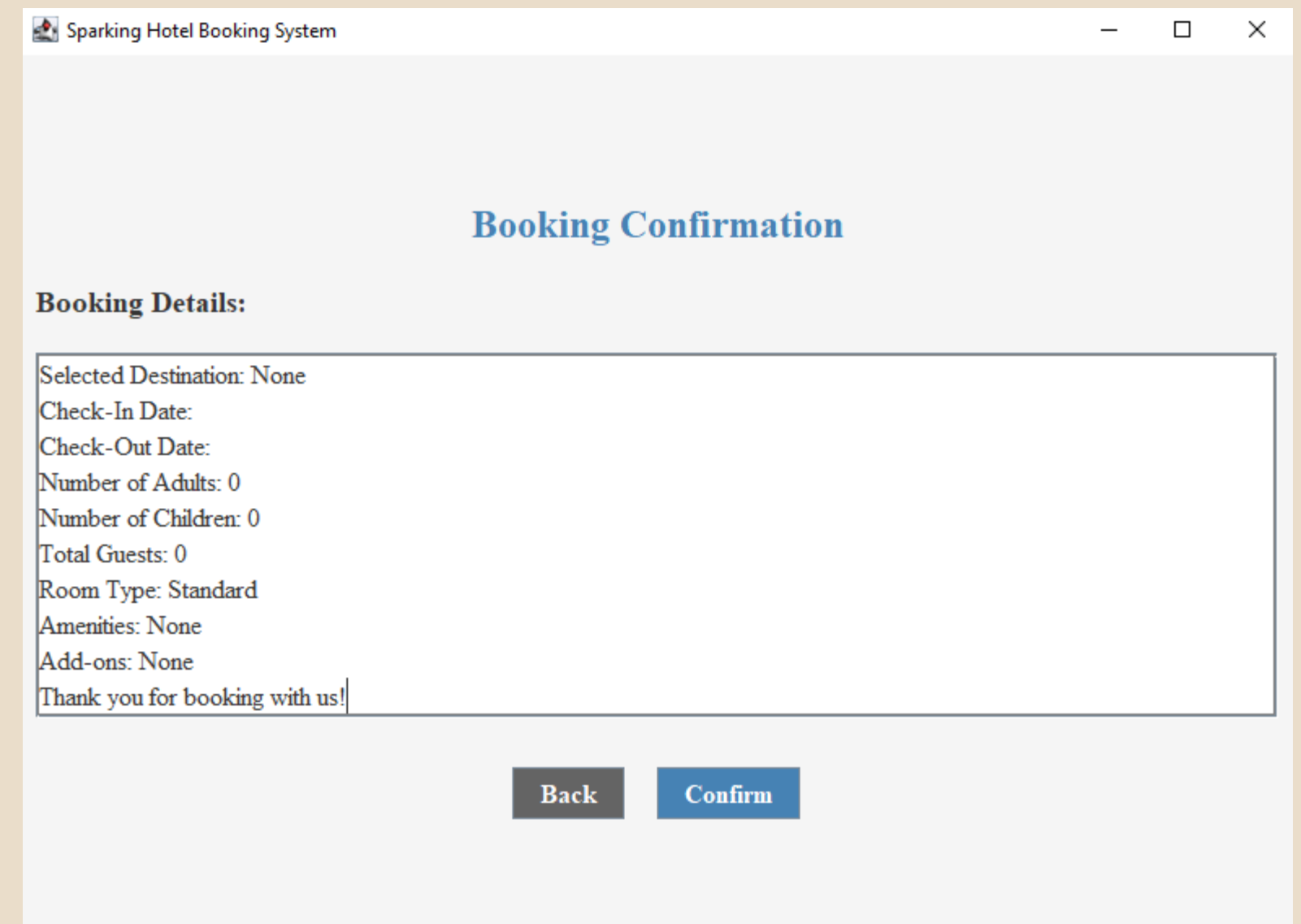
    // Room Types and Capacities
    String[][] roomDetails = {
        {"Standard", "Capacity: 1 guest with 1 extra bed"},
        {"Deluxe", "Capacity: 2 guests with 1 extra bed"},
        {"Quadruple", "Capacity: 4 guests with 1 extra bed"},
        {"Family", "Capacity: 6 guests with 1 extra bed"},
        {"Suite", "Capacity: 4 guests"}
    };

    for (String[] room : roomDetails) {
        JPanel roomPanel = new JPanel();
        roomPanel.setLayout(new BorderLayout());
        roomPanel.setBackground(new Color(245, 245, 245));
        roomPanel.setBorder(
            BorderFactory.createCompoundBorder(
                BorderFactory.createTitledBorder(room[0]), // Room type as border title
                BorderFactory.createEmptyBorder(5, 10, 5, 10) // Padding within the border
            )
        );
    }
}
```


USER TYPES & FEATURES

Features

- Automatically computes the total sum
- Reservation Summary



The screenshot displays a web application window titled "Sparking Hotel Booking System". The main heading is "Booking Confirmation" in blue. Below it, the section "Booking Details:" is followed by a list of booking information: "Selected Destination: None", "Check-In Date:", "Check-Out Date:", "Number of Adults: 0", "Number of Children: 0", "Total Guests: 0", "Room Type: Standard", "Amenities: None", and "Add-ons: None". At the bottom of the details box is the text "Thank you for booking with us!". Below the details box are two buttons: "Back" (dark grey) and "Confirm" (blue).

Sparking Hotel Booking System

Booking Confirmation

Booking Details:

Selected Destination: None
Check-In Date:
Check-Out Date:
Number of Adults: 0
Number of Children: 0
Total Guests: 0
Room Type: Standard
Amenities: None
Add-ons: None
Thank you for booking with us!

[Back](#) [Confirm](#)



THANK YOU
SO MUCH