



Software Design

Progress Report No. 7

---

## Designing the System

*Submitted by:*

**Group: BLDG.**

Gabijan, Rhovic M.

Balana, Jerkielle O.

Balaoro, Judge Wayne B.

Barbas, Steven Jade P.

Dispo, Lei Andrew T.

Laput, Mark Danielle E.

*Instructor:*

Engr. Maria Rizette H. Sayo

January 24, 2026

# I. Objectives

## Objectives

In this section, the goals in this laboratory are:

- To describe the conceptual design and the technical design of the system
- To transform the requirements into a working system

## II. Methods

### What is Design?

Our customers usually want a new system either because there is no existing system or because there are undesirable aspects of the old system. In either case, the requirements documents tell us all about the problem that the system is to solve. **Design** is the creative process of transforming a problem into a solution; the description of a solution is also called **design**.

### Two Part of the Design Process

**Conceptual or System design** - tells the customer exactly what the system will do

**Technical design** – that allows system builders to understand the actual hardware and software needed to solve the customer's problem

### Presentation Content:

When the design is complete, we meet with our customers to review it before development continues. The review process is done in three steps, corresponding to the steps of the design process. First, we hold a preliminary design review to examine the conceptual design with customers and users. Then, in a critical design review, we present the technical design to other developers to check its details before proceeding with implementation. Finally, we hold a program design review, so the programmers get feedback on their designs before implementation.

**Preliminary Design Review.** At a preliminary design review, we met with customers and users to validate the conceptual design. That is, we want to be sure that all aspects of the requirements are addressed by our design.

Instruction: Construct a preliminary design review of your proposed system. Describe the overall conceptual design to describe hardware configuration, software architecture, interface, the data structures, or you may use diagrams to show data flow, input, and output requirements. Discuss these all with your group.

## III. Results

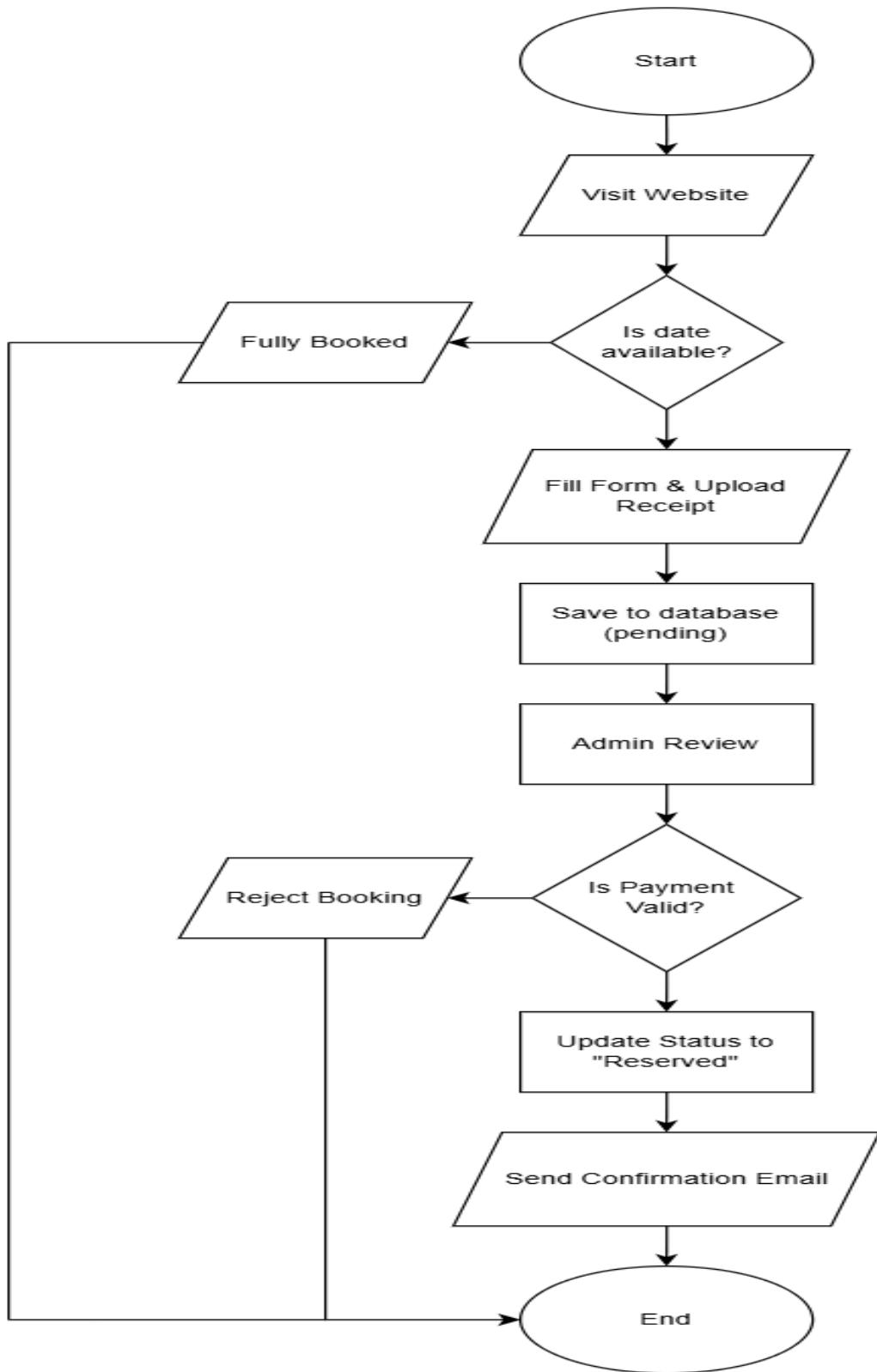
### System Overview

The proposed "Development of an Online Web-Based Booking System for Reservation and Resort Management" is designed to automate the resort's reservation process, transitioning from a manual logbook to a centralized digital database. The conceptual design follows a client-server model where the **Guest** interacts with a responsive frontend to check dates and submit payments, while the **Administrator** manages approval via a secure backend dashboard. This design directly addresses the client's need to eliminate double-bookings and streamline data retrieval.

### System Logic (Flowchart)

The process logic, as illustrated in the System Flowchart (**Figure 1**), follows a linear validation path:

1. **Guest Interaction:** The guest selects a date. If the system decision block "Is date available?" returns YES, the guest uploads their proof of payment.
2. **Admin Validation:** The system saves the booking as "Pending." The Administrator then manually reviews the receipt. If the decision "Is Payment Valid?" returns YES, the status is updated to "Reserved," and a confirmation email is sent.

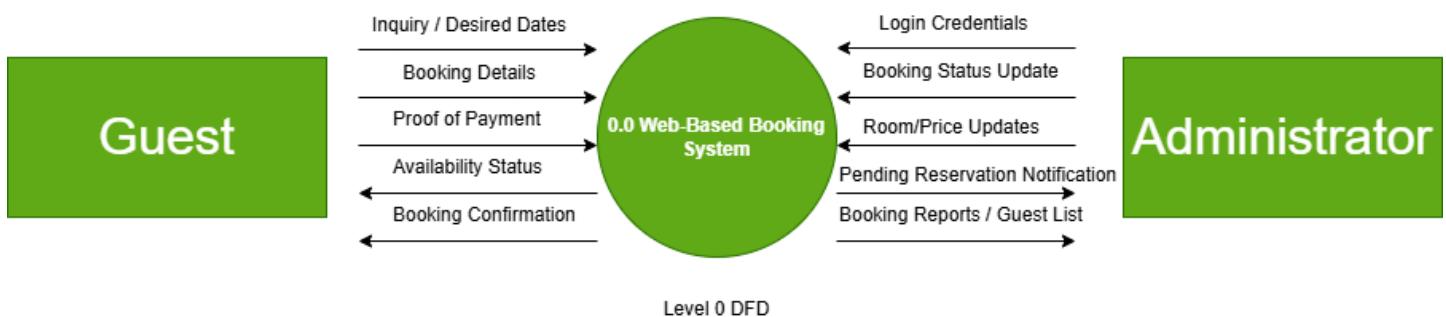


**Figure 1. System Flowchart of the Palacio Feliz Booking System**

## Data Flow Architecture (DFD Level 0)

To complement the logic flow, the **Data Flow Diagram Level 0** (Figure 2) illustrates the system's input/output requirements. The system acts as the central process (0.0) interacting with two external entities:

- **Guest Input:** Provides "Inquiry Dates" and "Proof of Payment."
- **Guest Output:** Receives "Availability Status" and "Booking Confirmation."
- **Admin Input:** Provides "Login Credentials" and "Booking Status Updates" (Approvals).
- **Admin Output:** Receives "Pending Reservation Notifications" and "Booking Reports."



**Figure 2. Context Diagram (DFD Level 0) showing system inputs and outputs**

## Hardware and Software Configuration

- **Software Architecture:** The system uses **Python** for the backend logic and **HTML5/CSS3** for the frontend.
- **Hardware Requirements:** The application is cloud-hosted requiring only standard devices (Laptop/Smartphone) with internet connectivity.

## IV. Conclusion

In this laboratory activity, the group successfully established the **Conceptual Design** for the Palacio Feliz Booking System. By mapping both the **logic flow** (Flowchart) and the **information flow** (DFD), we confirmed that the system effectively handles the client's data requirements. The design ensures that "Proof of Payment" is captured before any reservation is finalized, preventing double-bookings. This dual-diagram approach validates that the proposed architecture is robust enough for the upcoming technical implementation.

## **References**

- [1] Co Arthur O.. “University of Caloocan City Computer Engineering Department Honor Code,” UCC-CpE Departmental Policies, 2020.