**A PROPOSED OFFERING OF WEB-BASED RESERVATION SYSTEM FOR THE ADDLIB DANCE STUDIO AT DATAMEX COLLEGE OF SAINT ADELINE VALENZUELA BRANCH**

A Research Project Presented to the

Faculty of Datamex College of Saint Adeline, Inc.

In Partial Fulfillment of the Requirements for the

Degree of Bachelor of Science in Information Technology

Bacani, Princess Mahallalel

Delos Angeles, Arjay O.

Mejila, Steven Carl A.

Norte, Sakura Ann G.

**REQUIREMENTS**

**SPECIFICATION**

**INTRODUCTION**

Addlib Studio is a place where people can practice music, dance, and other creative activities. Many people go there to learn, practice, or rent the space for events. Right now, the way the studio handles reservations is mostly through walk-ins. Clients need to visit the studio personally just to book a class or a room. This process works, but it can be a hassle for both the clients and the staff.For clients, having to go all the way to the studio just to make a booking can be inconvenient. Sometimes they only want to check available schedules, but they still need to be physically present. For the staff, on the other hand, the system is also hard to manage. They need to record the bookings by hand and keep track of schedules, which can sometimes lead to mistakes or double bookings. Because of this, the current process can be stressful and time-consuming.

This project proposes the development of an Online Reservation System for Addlib Studio. The system will allow clients to make their bookings online, anytime and anywhere, without needing to visit the studio. By doing so, it gives more convenience to the clients since they can easily check available slots and reserve them with just a few clicks. It also reduces the chance of errors because the system can automatically prevent double bookings.The system will also have an admin dashboard that is designed for the studio staff. Through this dashboard, the admin can check all incoming reservations, confirm or decline requests, and monitor the schedule in real time. This feature is important because it helps the staff organize the bookings better and makes their work less heavy compared to manual recording.

Once a booking is confirmed, the system will automatically notify the client either through email or SMS. This is useful since it saves the staff from having to contact every client one by one. At the same time, clients can be sure that their reservation is confirmed without needing to follow up personally. This makes the process smoother for both sides.It is also important to note that Addlib Studio already has a website with sections like Home, About, Rental, Book a Class, and Contact Us. These pages provide basic information about the studio and its services. However, the booking process is still not automated, so the website cannot fully handle reservations yet.

The proposed system will improve this part by making the booking process faster and more efficient.The Online Reservation System for Addlib Studio will make a big improvement in how bookings are handled. Clients will no longer need to physically go to the studio, and staff will not have to deal with messy schedules. The system will save time, reduce mistakes, and make the overall experience more convenient for everyone. With this, Addlib Studio can give better service to its clients while making the work of the staff easier

**Overview**

The Online Reservation System for Addlib Studio is designed to improve the current way of handling bookings by moving from walk-in reservations to a digital platform. The system focuses on two main users: the clients and the administrators. For the clients, it provides an easier and faster way to check available schedules, reserve a slot, and receive confirmation without going to the studio. For the administrators, it offers a management dashboard where all reservations can be tracked and controlled in real time.

The system also adds important features that make the process more reliable. These include automatic notifications through email or SMS, prevention of double bookings, and mobile access so that clients can make reservations anytime using their phones. At the same time, admins gain access to reports that can help in analyzing booking trends and making better business decisions.

The project aims to make the reservation process more convenient for clients while reducing the workload of staff. By using this system, Addlib Studio can improve its services, operate more efficiently, and provide a better experience for everyone who wants to book classes or rent the studio

**PROJECT SCOPE**

The proposed system will be a secure, web-based platform where clients can reserve classes or rent the studio online. One of its main features is that it will help avoid double bookings by showing which schedules are already taken. This way, clients can clearly see what times are open before they make a reservation.The platform will also be mobile-friendly, which means clients can use their phones or laptops when making reservations. Aside from booking, they will also be able to cancel if they change their plans, or even join a wait-list if the schedule is already full.

This gives more flexibility for clients and makes the process less stressful.For the admin side, the system will provide tools for managing schedules and handling booking requests. Admins can view all the reservations in one place, then confirm or decline them based on availability. This will help them avoid errors that usually happen when everything is done manually.The owners will have access to reports that summarize the studio’s reservations. This will allow them to check how the system is being used and also see patterns, like which times are the busiest. Having this kind of overview helps in making better decisions for the studio in the future.Notifications are also included in the system. Once an admin confirms or declines a booking, the client will automatically get a message through email or SMS. Reminders can also be sent out to make sure clients are aware of their reservations. This makes communication between the studio and clients clearer and more reliable

.

Inclusion, the project’s goal is to make studio operations easier to manage while also giving clients a better booking experience. It reduces the workload of staff, avoids scheduling problems, and gives clients a faster way to reserve classes. In the long run, the system is expected to improve both efficiency and customer satisfaction.

**FUNCTIONAL REQUIREMENTS**

This document outlines the functional requirements for the online reservation system. It details the features necessary to ensure smooth interaction between clients and administrators, covering booking, scheduling, notifications, and conflict prevention. Each requirement is listed with its priority, dependencies, and acceptance criteria to provide clear guidance for development and validation.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Requirement ID** | **Requirement Description** | **Priority** | **Dependencies** | **Acceptance Criteria** |
| **REQ-001** | Clients can create online reservations through a web page. | **High** | **None** | Clients can open the reservation form, fill details, and submit successfully**.** |
| **REQ-002** | Clients can select date, time, and service when booking. | **High** | **REQ-001** | Form shows available slots, avoids overlaps, and only accepts valid input. |
| **REQ-003** | The system sends a booking request notification to admin dashboard. | **High** | **REQ-001** | Reservation appears immediately on the admin dashboard |
| **REQ-004** | Admin can view, confirm, or decline reservations on dashboar**d.** | **High** | **REQ-003** | Admin can approve or reject bookings successfully**.** |
| **REQ-005** | Clients get notified by email or SMS when booking is confirmed or declined. | **High** | **REQ-004** | Notifications are received within 1 day. |
| **REQ-006** | System prevents overlapping or double bookings. | **High** | **REQ-002** | No two clients can reserve the same slot. |
| **REQ-007** | Reservation details are stored securely in a database. | **High** | **REQ-001** | Database keeps complete booking records with timestamps**.** |
| **REQ-008** | Admin can view analytics (e.g., peak hours, bookings per day). | **Low** | **REQ-004, REQ-007** | Dashboard shows reports or graphs with booking stats**.** |

Table 1. “The Functional Requirements”

**NON-FUNTIONAL REQUIREMENTS**

**Performance**

* Pages should load within 3 seconds.
* Reservation submissions update in 2 seconds or less.
* Supports at least 100 users at the same time.

**Usability**

* System must be user-friendly and easy to navigate.
* Works on both computers and mobile devices.

**Reliability**

* Should always be available for use.
* No data should be lost due to errors or crashes.
* Daily backups will be done to secure data.

**Security**

* Only admins and head-admins can log in to dashboards with secure credentials.

**Scalability**

* Can handle up to 1000 reservations if needed.

**Maintainability**

* Updates and bug fixes should be easy to apply without long downtime.

**USE CASES**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Use Case ID | Use Case Name | Description | Actors | Preconditions | Postconditions | Alternate Flows |
| Phase 1 | Online Reservation Submission | Client selects service, date, and time, then submits booking. | Client, System | Client has internet access; system is working. | Booking saved in database; admin notified. | If slot is taken → client must choose another. |
| Phase 2 | Manage Reservation Requests | Admin checks pending reservations and confirms/declines. | Admin, System | Admin is logged in. | Reservation status updated; client notified. | Session timeout → re-login required. |
| Phase 3 | Reservation Confirmation Notification | System sends notification when booking is approved. | Client, System | Reservation exists; admin approved. | Client gets confirmation database updated. | If message fails → system retries. |

Table 2. “The Use Cases”

**DATA REQUIREMENTS**

* Client: ClientID, Name, Email, Phone, Password (1 client can have many reservations).
* Reservation: ReservationID, ClientID, ServiceID, Date, Time, Status, CreatedAt, UpdatedAt, ConfirmedBy.
* Service: ServiceID, Name, Description, Duration, Price (1 service can have many reservations).
* Admin: AdminID, Name, Email, Password, Role (sets schedules).
* HeadAdmin: HeadAdminID, Name, Email, Password, Role (confirms/declines reservations).
* Schedule: ScheduleID, ServiceID, AdminID, Date, StartTime, EndTime, MaxClients.

**ASSUMPTION & CONSTRAINS**

Clients have access to the internet and mobile devices

It is assumed that all clients interacting with the system will have reliable internet connectivity and access to devices such as smartphones, tablets, or computers. This assumption is crucial because the system relies heavily on online bookings, notifications, and potentially online payments. Without internet access, clients may be unable to make or manage bookings, which could affect the overall efficiency and effectiveness of the system.

Administrators will regularly check and confirm bookings

The system presumes that administrators or staff responsible for managing bookings will routinely monitor incoming requests and confirmations. Regular oversight ensures that scheduling conflicts are minimized, client requests are handled in a timely manner, and overall service quality is maintained. A lack of timely checks could result in double bookings, missed appointments, or client dissatisfaction.

Staff will input accurate schedules and availability

The accuracy of the system is highly dependent on staff entering correct information regarding schedules, availability, and service times. Incorrect or outdated entries could lead to booking errors, client confusion, and operational inefficiencies. This assumption places responsibility on staff to maintain updated records to ensure the system functions as intended.

Clients will provide accurate contact details

For notifications, reminders, and follow-up communication, it is assumed that clients will supply correct and current contact information. This includes phone numbers, email addresses, and any other relevant contact channels. Incorrect details could result in missed notifications, unconfirmed bookings, or clients arriving at the wrong time, impacting both client experience and operational efficiency.

No client dashboard for tracking schedules and attendance

The current system does not feature a client-facing dashboard that allows users to view their personal schedules, attendance history, or upcoming appointments. This limitation may affect client engagement and convenience, as they cannot independently verify their bookings or track their participation. The absence of this functionality may also increase reliance on administrators for schedule-related queries, adding to their workload.

Dependence on manual checks and confirmations

Since the system assumes that administrators will manually check and confirm bookings, there is a constraint on scalability. As the number of clients grows, relying solely on manual oversight could lead to delays or errors unless additional automation or administrative support is implemented in the future.

Reliability on staff for data accuracy

Another constraint is the system’s reliance on staff to input correct schedules and availability. Human error is always a possibility, which can lead to scheduling conflicts, double bookings, or misinformation being sent to clients. This constraint emphasizes the need for proper training and possibly validation mechanisms to reduce errors.

**GLOSSARY**

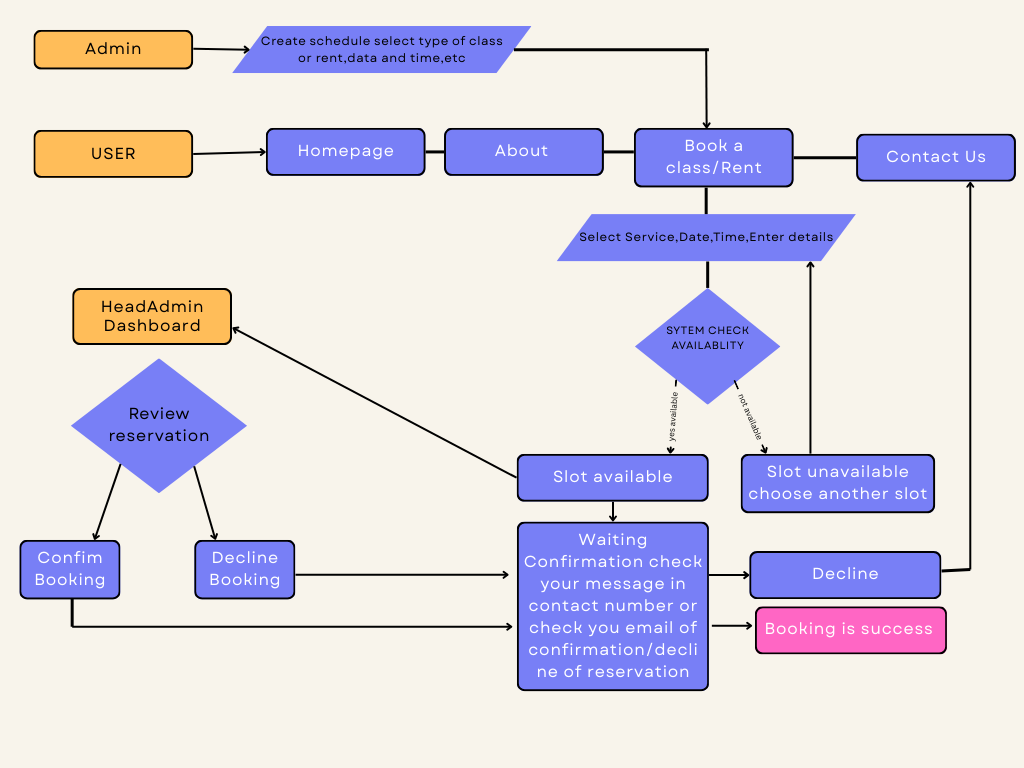
* Head Admin: Person who has the highest control and can confirm/decline bookings.
* Admin: Person who sets class schedules.
* Client: User booking a class or studio rental.
* Reservation Dashboard: Where admins manage reservations.

**REVISION HISTORY**

|  |  |  |
| --- | --- | --- |
| Timeline | Description | Changes |
| 08/05/2025 | Developing an Online Reservation and Management System for Addlib Dance Studio | Updated formatting, removed user dashboard, and designated SQL Server as the database. |

Table 3. “Revision History”

**APPENDIX**

****

* **Step 1: Admin Sets Up Schedule**

The admin logs into the system and creates the studio’s schedule. They choose which classes or services will be available, set the date and time for each, and decide how many clients can join. This step is important because it tells the system which slots clients can book.

* **Step 2: Client Visits Website**

The client opens the Addlib Studio website. They can look at pages like Home, About, and Contact Us to know more about the studio. When they want to book a class or rent a service, they go to the “Book a Class” or “Rent a Service” page.

* **Step 3: Client Chooses Service and Fills Details**

On the booking page, the client picks the class or service they want. They select a date and time that fits their schedule and fill in personal information like their name, email, and phone number.

* **Step 4: System Checks Availability**

After the client submits the form, the system checks if the chosen slot is free. If the slot is already taken, the client is asked to choose a different time. If there are no available slots, the system may decline the booking.

* **Step 5: Booking Sent to Head Admin (If Needed)**

If the slot is free, the booking request goes to the Head Admin dashboard. The Head Admin checks all the details to make sure everything is correct.

* **Step 6: Head Admin Confirms or Declined**

The Head Admin decides if the booking can be approved or not. They can either confirm the reservation or decline it. This step helps keep the schedule organized and prevents mistakes.

* **Step 7: Client Gets Notification**

Once the Head Admin makes a decision, the system sends a message to the client by email or SMS. If the booking is approved, the client gets a confirmation. If it is declined, the client is told that the booking could not be made.

* **Step 8: Booking Completed**

The system records the booking. If it was confirmed, the client’s reservation is added to the schedule. If it was declined, the system updates the status. This keeps the schedule clear and lets both the staff and the clients know the result