**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.

**DEPLOYMENT DOCUMENTATION**

# **INTRODUCTION**

The Addlib Dance Studio Booking System is a web-based application made to help people easily book dance classes, private lessons, and studio rentals online. Before this system was made, most students and clients of the dance studio had to book their classes through text messages or by visiting the studio in person. This process often caused confusion, double bookings, and mistakes in the schedule. The new system helps to solve those problems by giving a simple and organized way to handle all bookings through one website.

This booking system is designed not only for customers but also for the dance studio staff. Customers can open the website, look at available classes, and make a reservation quickly. They can also see the details of their bookings and cancel if they need to. On the other hand, the staff can easily manage all bookings in one place. They can see which classes are full, which times are available, and who has booked each session. This saves a lot of time and makes their work easier.

The main goal of this deployment is to make the booking system available online so that real users can start using it in a live environment. The system’s frontend, or the part that users see, is hosted on GitHub Pages, while the backend, which stores and manages all the data, is hosted using Supabase. Supabase provides a secure database, handles user accounts, and stores all the booking information safely. This setup allows the system to work properly and be accessible to anyone with an internet connection.

The scope of this deployment includes setting up the system for real use, connecting the frontend to the Supabase backend, and checking that all features work correctly. It also includes testing, staff training, and support after the system goes live. Once deployed, this project will help Addlib Dance Studio become more modern and efficient. Clients will have an easier time booking classes, and the staff will have a smoother way to manage their schedules. Overall, the system aims to improve communication, save time, and make the booking experience simple and convenient for everyone..

**DEPLOYEMENT PLAN**

The deployment environment is the setup where the Addlib Dance Studio Booking System will run and be accessed by users. It includes all the hardware, software, and hosting tools needed to make the system work properly. Having the right environment is important to ensure the website performs smoothly and can handle all the booking and scheduling tasks. The environment also needs to be secure and reliable so that user information and booking records are protected at all times.

For the hardware requirements, the system only needs basic computer equipment since it is a web-based application. A computer or laptop with a stable internet connection is enough to access and manage the system. The server used for hosting does not require physical setup because the system is deployed online. Users can access the website through any device such as a desktop, laptop, tablet, or even a mobile phone. A good internet connection is the most important requirement to ensure smooth and fast access to the system.

In terms of software, the system uses GitHub Pages for hosting the frontend part and Supabase for the backend database and authentication. The frontend is built using modern web technologies such as HTML, CSS, and JavaScript, while the backend in Supabase handles user accounts, booking records, and class data. Supabase provides a reliable and secure cloud-based database that automatically updates in real time. The hosting environment allows the dance studio’s booking system to be available anytime and anywhere, making it more convenient for both the staff and the clients. important because it ensures that the system continues to perform well even after it goes live

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Phase | Description | Start Date | End Date | Status |
| Pre-Deployment | Preparing the environment and testing the system | Oct10,2025 | Oct15,2025 | Completed |
| Deployment | Uploading the system and settings | Oct10,2025 | Oct20,2025 | In Progress |
| Post-Deployment | Monitoring,Testing and training | Oct10,2025 | Oct30,2025 | Pending |

**DEPLOYMENT ENVIRONMENT**

The deployment environment describes all the tools, systems, and services needed to make the Addlib Dance Studio Booking System run properly. Since this system is web-based, it doesn’t need heavy hardware or local servers. It mostly depends on cloud hosting and internet access to work smoothly. This section explains the hardware, software, and hosting setup used for deployment

Hardware Requirements:

Frontend: Hosted online using GitHub Pages, so no physical hardware is required.

Backend: Uses Supabase, which is a cloud-hosted platform built on PostgreSQL.

Client Devices: Any device that can access the internet such as desktops, laptops, tablets, or smartphones.

Browser Requirements: Works best with modern browsers like Google Chrome, Mozilla Firefox, Safari, or Microsoft Edge.

Network Requirements: Needs a stable internet connection for fast loading and smooth booking transactions.

Software Requirements:

Frontend Stack: Built using HTML, CSS, and JavaScript (can be React or Vanilla JS).

Backend Stack: Managed through Supabase

APIS and Depencies:

Supabase SDK

Environment variables for VITE\_SUPABASE\_URL and VITE\_SUPABASE\_ANON\_KEY

Hosting Information:

Frontend Hosting: GitHub Pages (repository: surugiii/dance-studio-booking).

Backend Hosting: Supabase (cloud-managed).

Domain: <https://surugiii.github.io/dance-studio-booking/>

Database: Supabase-managed PostgreSQL database.

Authentication: Supabase Auth (email/password or social login).

Storage: Supabase Storage for saving uploaded files (if used).

**DEPLOYMENT PROCEDURES**

The Addlib Dance Studio Booking System has been successfully deployed and is now available online. This means that users can access the system from anywhere using a web browser, without needing to install any special software. The deployment makes it easier for dance studio staff and clients to view class schedules, make bookings, and manage reservations all in one place.

**PRE-DEPLOYMENT STEPS**

1.Backup existing data (if applicable): Ensure any old records or previous bookings are saved safely before deployment, to prevent accidental data loss.

2.Set up the required environment: Install necessary software, configure the database on Supabase, and make sure the frontend and backend can communicate correctly.

3.Check network connectivity and system compatibility

4.Test the local environment: Run the system locally to confirm that login, booking creation, and schedule display work properly before going liv**e.**

**DEPLOYMENT EXECUTION**

1.Deploy the system online

Host the frontend on GitHub Pages.

Use Supabase for the backend to manage the database and user authentication.

Share the live system link: https://surugiii.github.io/dance-studio-booking/

2.Configure system settings

Make sure all environment variables are correct.

Check database connections and API keys.

Confirm that the frontend and backend communicate properly.

3.Initialize the system

Open the application online and go through all features.

Test login, booking creation, and schedule viewing.

Make sure everything runs smoothly and no errors appear.

4.Perform initial testing

Test all major features in the live environment. And Identify any bugs or issues.

**POST DEPLOYMENT STEPS**

1.Verify functionality

Run test cases to confirm the system works for staff and clients.

Test logging in, booking classes, and viewing schedules.

2.Monitor system performance and stability

Regularly check Supabase dashboards and GitHub Pages.

Look for errors, slow responses, or connection issues.

3.Conduct user training

Teach staff how to manage bookings, add or remove classes, and monitor client data.

Provide guidance for clients on using the system effectively.

4.Provide ongoing support

Share contact information for reporting issues or asking questions (email and phone).

Encourage users to report bugs or problems they encounter.

Maintain a smooth and user-friendly experience for everyone

**USER TRAINING & SUPPORT**

Training and support are very important to make sure that both the staff and clients can use the Addlib Dance Studio Booking System easily and without confusion. Before the system goes live, the staff take part in pre-deployment training sessions. These sessions teach them how to use the admin dashboard, manage class schedules, handle bookings, and assist clients if any issues come up. The goal is to make sure that staff feel confident in using the system so that daily operations at the dance studio run smoothly.

After the system is deployed, clients are also guided on how to use it. They receive a walkthrough or tutorial, which can be a video or a simple guide. This helps clients understand how to log in, search for available classes, make bookings, and check their schedules. The walkthrough is designed to be easy to follow, so even people who are not very tech-savvy can use the system without difficulty.

In addition to live training and walkthroughs, clear documentation is provided for both staff and clients. Clients receive a user manual that explains how to browse class offerings, make reservations, log in to their accounts, and contact support if needed. Staff receive an administrator manual that shows how to set up classes, manage instructors, handle bookings, and generate reports. There are also quick reference guides and FAQ documents to help both staff and clients solve common problems quickly, without needing to wait for support.

Ongoing support is available to make sure that users always have help when they need it. Users can contact the support team through email at studio\_support@addlibstudio.ph or by phone at +68992340293049. Support is available during weekdays from 9:00 AM to 6:00 PM, and emergency support is available in the evenings and on weekends for urgent issues. The support team also uses a ticketing system, such as Jira or ServiceNow, to track reported problems and ensure that every issue is addressed quickly. By providing structured training, clear guides, and continuous support, both staff and clients can use the system efficiently and confidently.

**RISKS & CONTINGENCY PLAN**

Deployment can face several risks, so it is important to identify potential issues and plan mitigation strategies. The following table summarizes the main risks for the system:

|  |  |  |
| --- | --- | --- |
| Risk | Impact | Mitigation Strategy |
| Server downtime | High | Ensure backup servers are available and notify users in advance. |
| Database connection failure | Medium | Test database connectivity before deployment. |
| User resistance | Low | Provide training and support to help users adapt. |
| Hacking / Security | High | Use strong passwords, secure API keys, and Supabase security rules. Regularly monitor logs for suspicious activity and apply updates or patches |

Deploying the Addlib Dance Studio Booking System involves some risks that need to be managed. The main risks include server downtime, database connection failure, user resistance, and hacking or security breaches. To reduce these risks, backup servers are used, database connections are tested, training and support are provided for users, and strong security measures are applied, such as secure passwords, API keys, and monitoring for suspicious activity. Planning for these risks helps ensure the system is safe, reliable, and easy to use for both staff and clients.

.

**DEPLOYMENT VERIFICATION & SIGN-OFF**

After deployment, the system undergoes verification to ensure that it functions correctly. This includes testing authentication, booking creation, schedule viewing, and overall performance on both taff and client sides. All tests must pass to confirm the system is ready for real users.

Stakeholder sign-off is required to officially confirm the deployment completion. This ensures that all responsible parties have reviewed the system, verified its functionality, and agreed that it is ready for operational use

|  |  |  |  |
| --- | --- | --- | --- |
| Stakeholder | Role | Signature | Date |
| Bacani,Princes Mahallalel | Project Manager |  | October 25, 2025 |
| Joe Abuda | Client Representative |  | October 25, 2025 |