OUR LADY OF FATIMA UNIVERSITY
COLLEGE OF COMPUTER STUDIES
ANTIPOLO CITY

A Web and Mobile Based Queueing Management System for Our Lady of Fatima University Antipolo Registrar Office

SYSTEMS ADMINISTRATION AND MAINTENANCE W/ LAB – Maintenance Guide

Presented to the Faculty of the

College of Computer Studies

OUR LADY OF FATIMA UNIVERSITY

Antipolo City

by:

LEBRINO, ERWIN

AGUSTIN, SEAN ARIEL H.

URSAL, JOHN MICHAEL V.

REPASO, JOHN ROMMEL S.

In partial Fulfillment of the Requirements for the Degree

BACHELOR OF SCIENCE IN INFORMATION TECHNOLOGY

Queueing Management System Maintenance's Guide

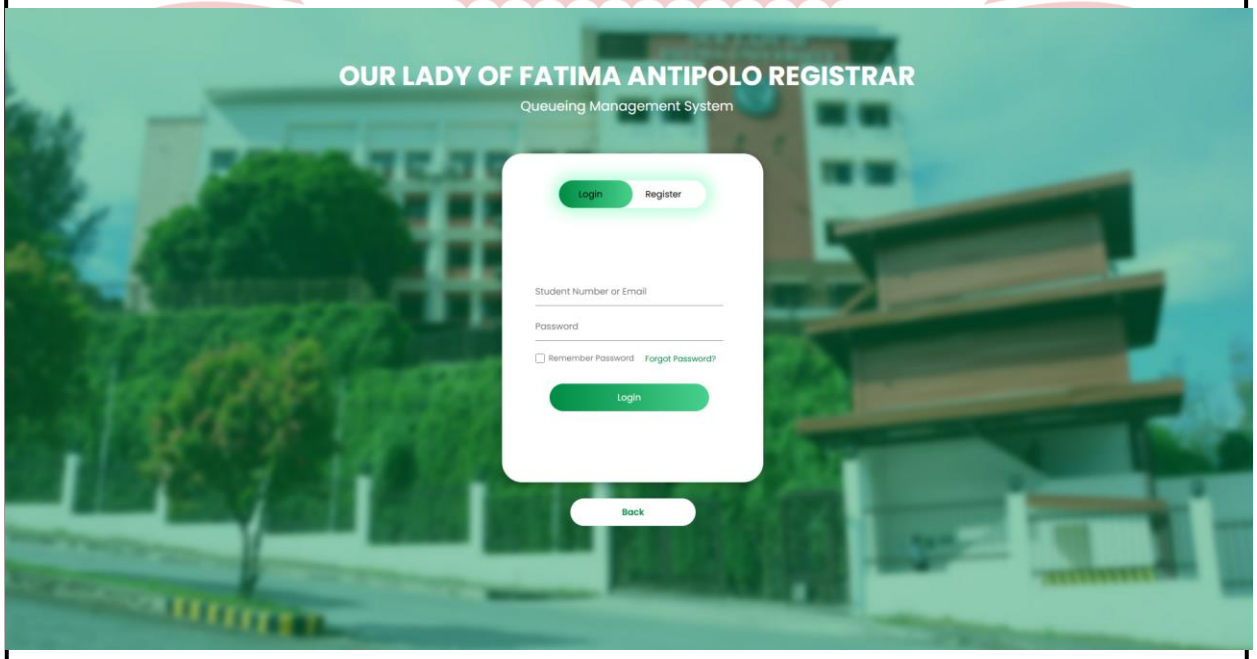
Chapter 1: Getting Started

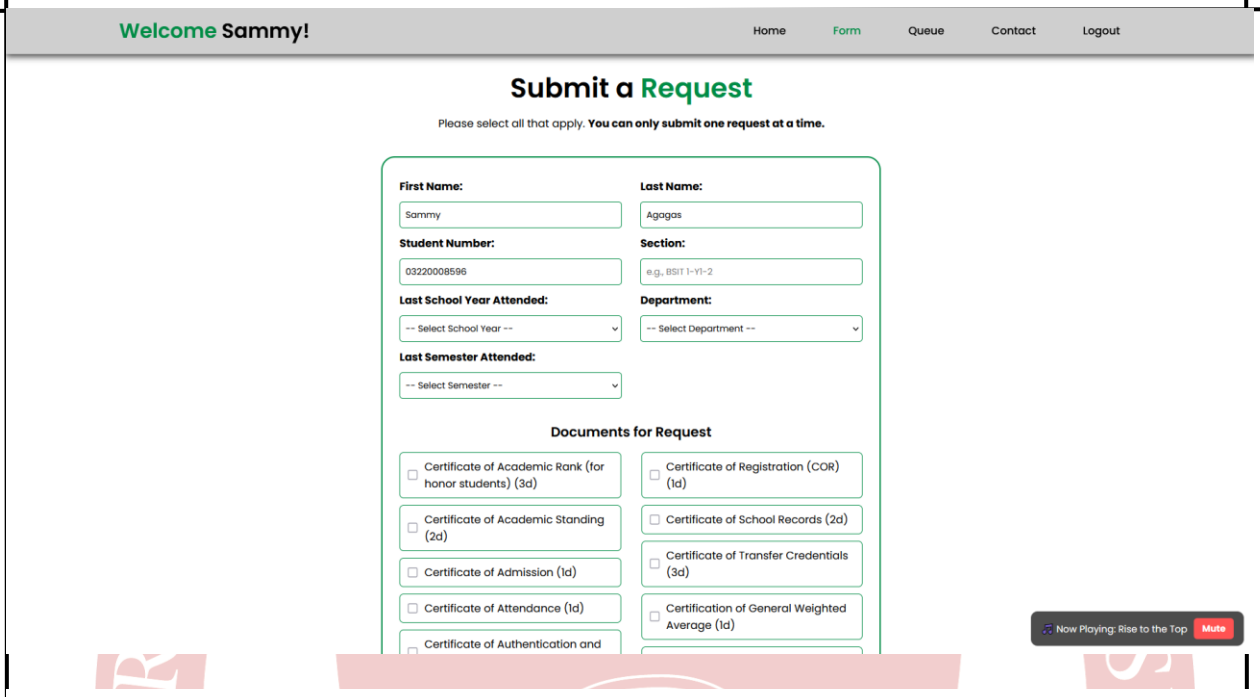
1.1 Introduction

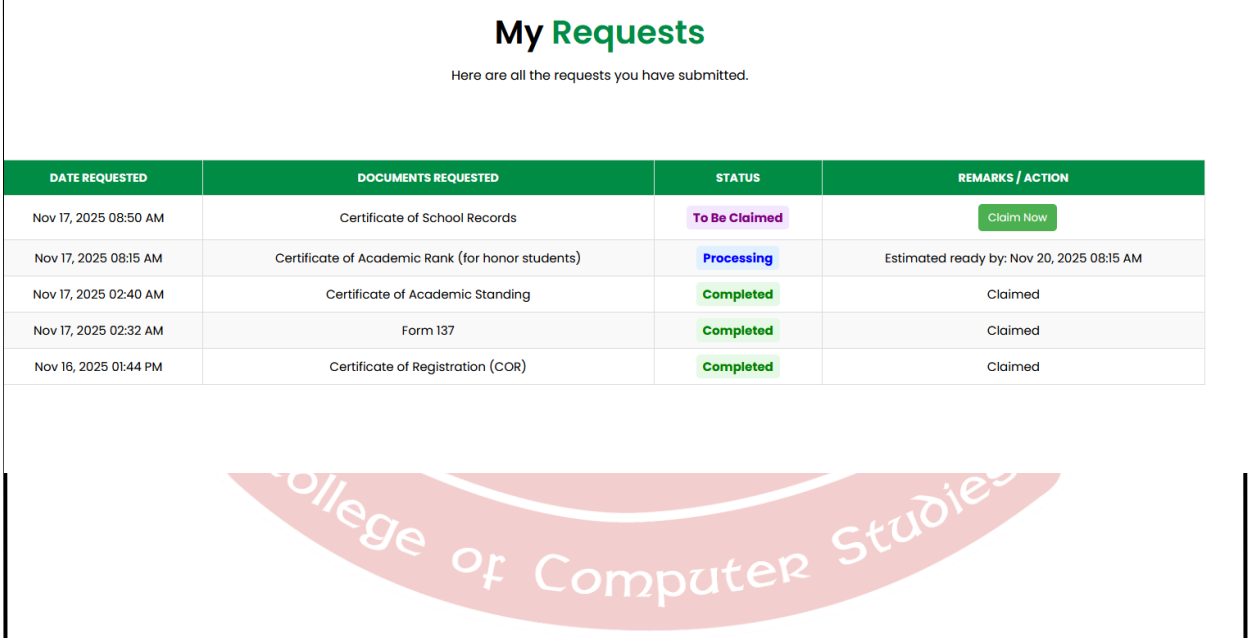
The Web-Based Queueing Management System of OLFU Antipolo facilitates efficient student interactions with the Registrar. It reduces waiting times, improves service organization, and provides a reliable method for managing student queues. The system also sends notifications to students when their queue is near, ensuring orderly processing.

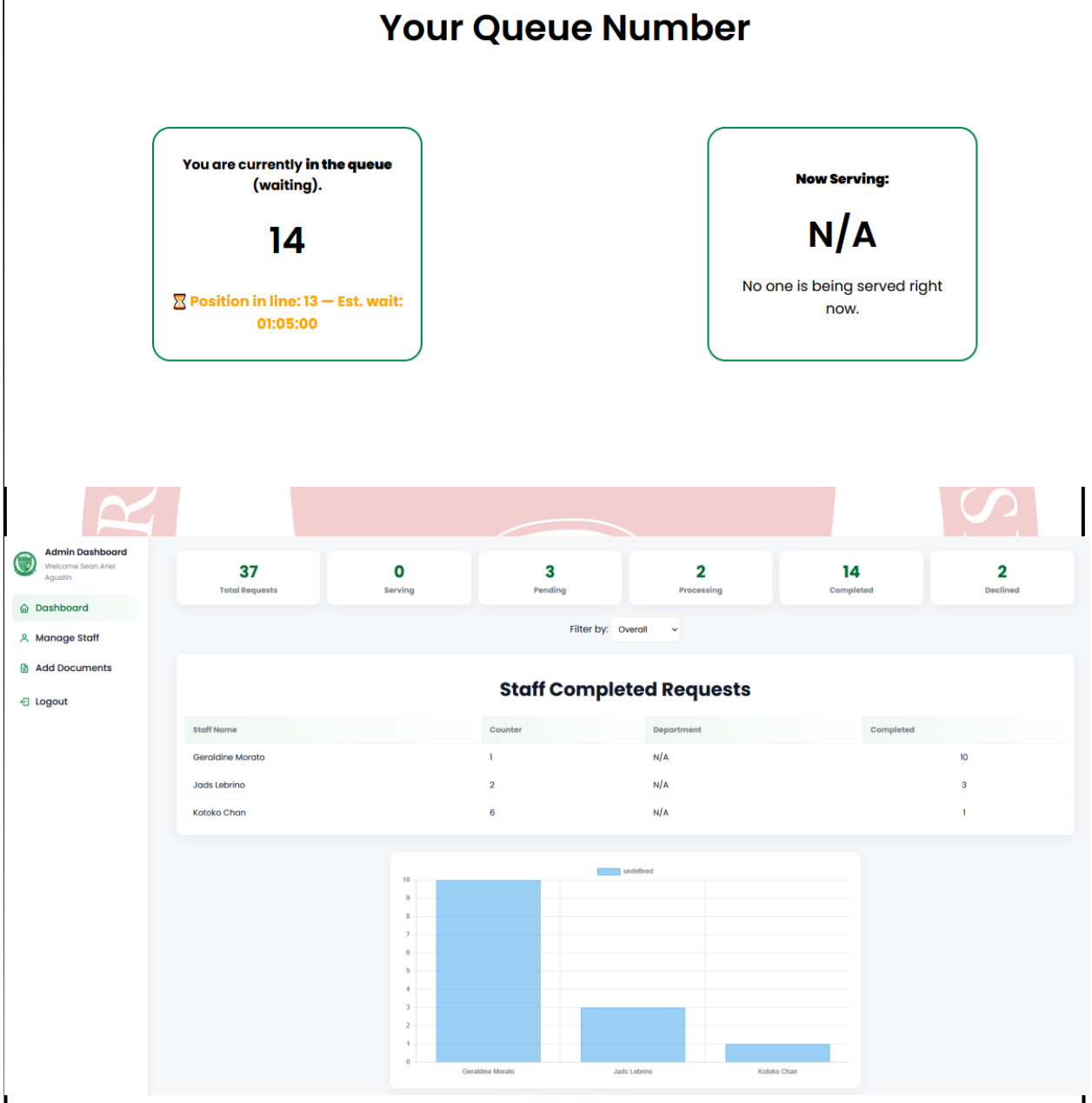
1.2 System Overview

The system consists of a **student-facing interface** and an **administrative backend**. The student interface allows selection of service types such as Document Requests, Clearance, and Records. The administrative dashboard enables staff to manage queues in real-time, view requests, and update statuses. The system is mobile-responsive, allowing access via smartphones, tablets, and desktop browsers.









1.3 Maintenance Objectives

The primary objectives of maintenance are to ensure **system uptime**, **data integrity**, **accurate notifications**, and **optimal performance**. Maintenance also involves addressing software bugs,

applying timely updates, monitoring resource usage, and ensuring the system can handle peak queue volumes during enrollment periods or end-of-semester periods.

1.4 Maintenance Team Roles & Responsibilities

- **System Administrator**: Monitors server performance, handles backups, ensures hosting reliability.
- **Developers**: Address bugs, deploy updates, maintain source code, and implement system enhancements.
- **Registrar Staff**: Report issues, verify queue accuracy, and assist students in using the system.

  Clear coordination among these roles ensures uninterrupted services and effective problem resolution.

1.5 System Architecture Overview

The system uses a **web-based frontend** for students and a **backend dashboard** for staff. The frontend handles user inputs, queue reservations, and notifications. The backend processes queue logic, stores records in a database, and triggers notifications. The modular architecture allows separate maintenance of frontend, backend, and database components without affecting the other modules.

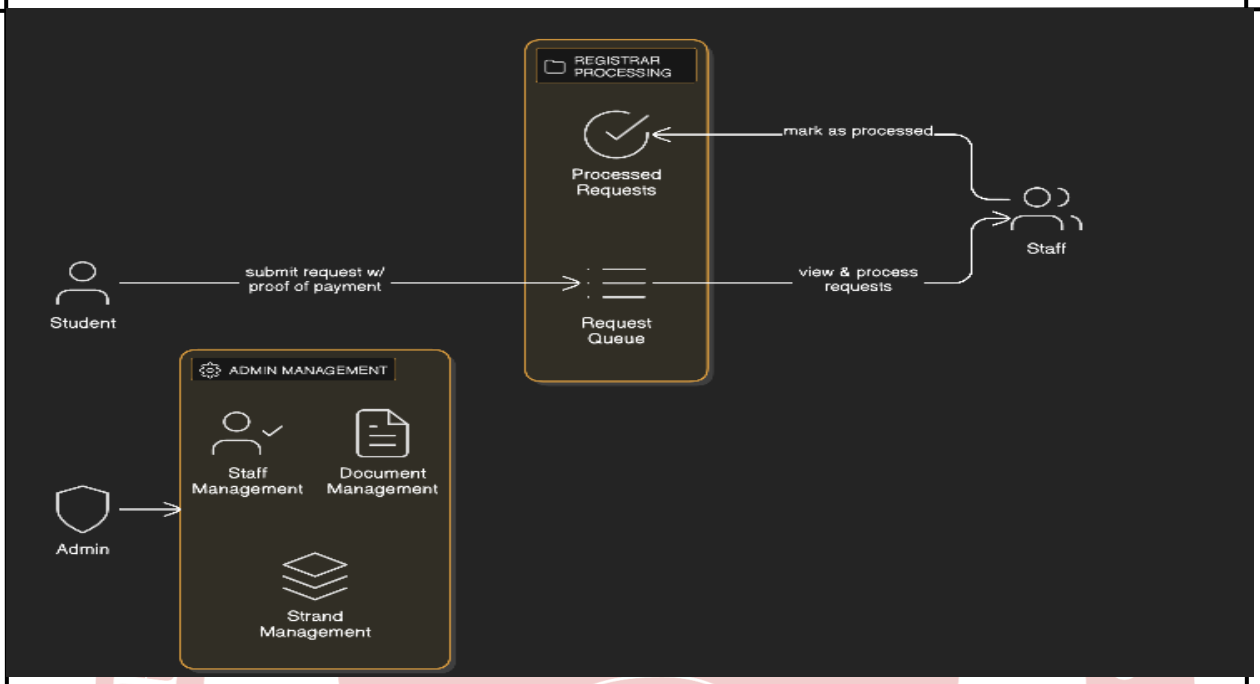


Figure 1.1 System Architecture Diagram

1.6 Technology Stack Summary

Although specific technologies are not listed on the website, the system employs standard web application components:

- **Frontend:** Responsive web design (HTML, CSS, JavaScript) for cross-platform accessibility.
- **Backend:** Server-side scripting to handle queue logic and administrative functions.
- **Database:** Stores user and queue data; optimized for fast query processing.
- **Security Protocols:** Role-based access control, encrypted connections via SSL.

Each component is selected to ensure reliability, performance, and scalability.

1.7 Server & Hosting Overview

The system is hosted at https://olfu-registrar.ellequin.com/. Hosting maintenance includes monitoring uptime, SSL certificate validity, server logs, storage usage, and ensuring compatibility with server-side scripts. Periodic checks ensure that hosting issues do not disrupt user access. Key hosting maintenance tasks include:

1. **Uptime Monitoring:** Check server availability using automated monitoring tools.
2. **SSL Validation:** Verify secure HTTPS connection for all users.
3. **Storage Usage:** Monitor disk space to prevent downtime due to capacity issues.
4. **Log Reviews:** Inspect server logs for errors or anomalies.
5. **Software Compatibility:** Ensure server-side scripts are compatible with the hosting environment.

Periodic reports should be generated and reviewed to preemptively detect potential issues.

1.8 Hardware & Network Maintenance Requirements

Even though the system is web-hosted, staff rely on local hardware and network stability. Key maintenance tasks include:

- **Workstation Checks:** Ensure staff computers are operational, free of malware, and meet minimum performance requirements.

- **Network Verification:** Maintain stable internet connectivity; verify VPNs or firewalls do not block access.
- **Peripheral Devices:** Printers and scanners used for queue reports must be functional and properly connected.

Regular verification of hardware and network ensures that staff can consistently access and manage the system.

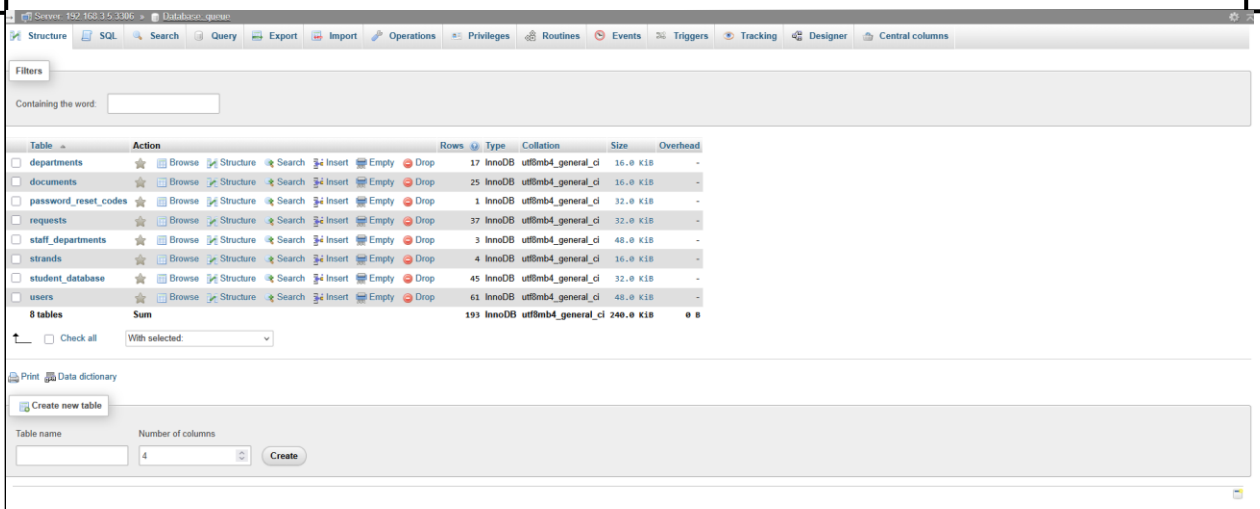
Chapter 2: Core Maintenance Activities

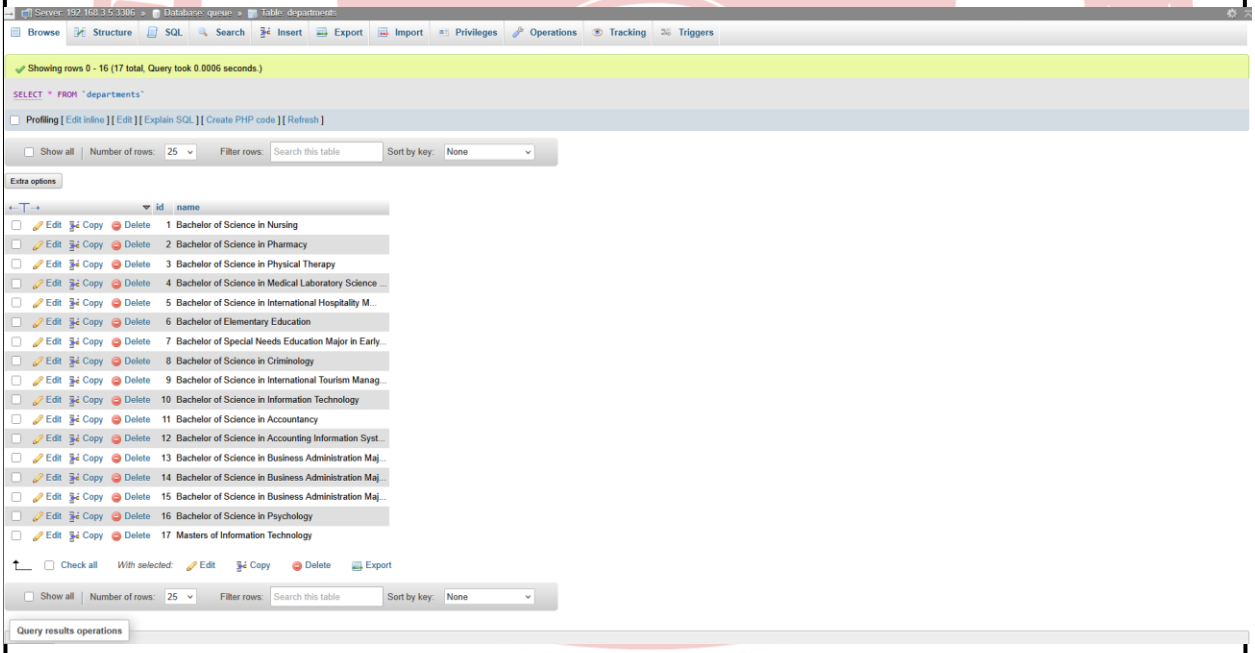
2.1 Database Maintenance Procedures

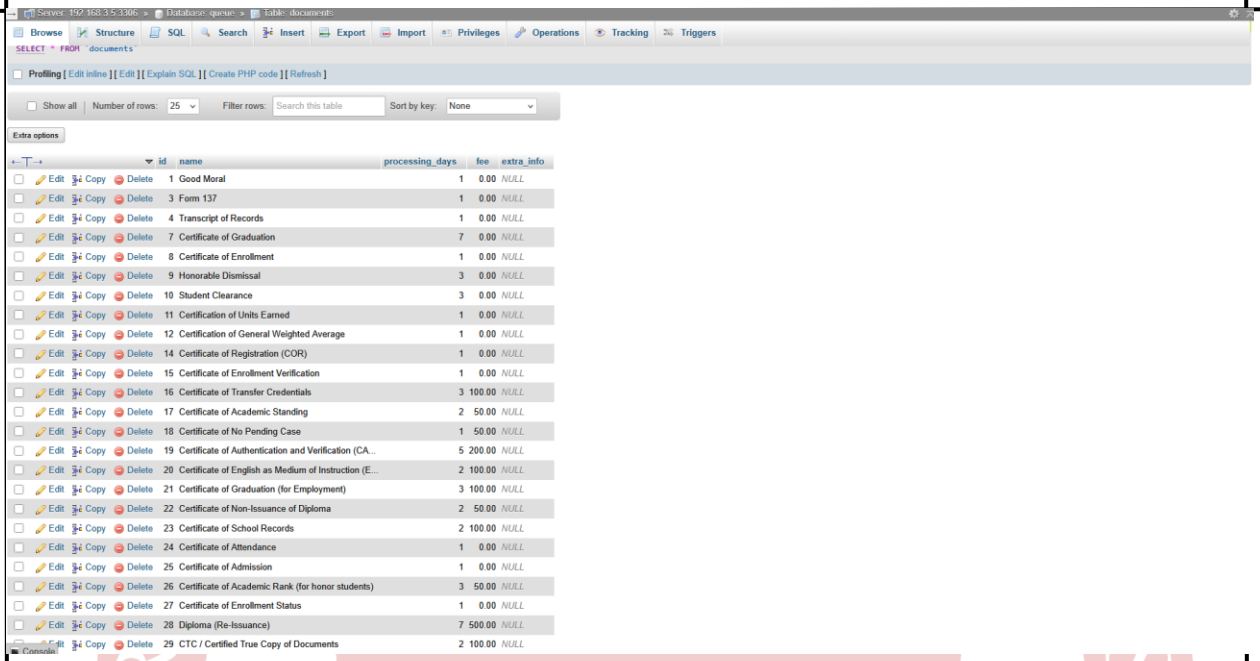
Verify table integrity, check for errors, optimize queries, and periodically archive outdated queue records. Maintenance involves monitoring table sizes, verifying correct data insertion, optimizing queries, and archiving old queue records. Regular database checks prevent performance degradation and ensure that queues are processed accurately.

Step-by-Step Database Maintenance:

1. **Check Table Integrity:**
    - Use SQL commands or database management tools (e.g., phpMyAdmin) to verify tables are free from corruption.
    - Ensure primary keys and relationships are intact.
2. **Validate Queue Records:**
    - Confirm new queue entries are stored correctly.
    - Sample query: SELECT * FROM queues WHERE status='pending';
3. **Optimize Queries:**
    - Index frequently accessed columns, e.g., queue_number, student_id, service_type.
    - Remove unnecessary or redundant indexes to improve efficiency.
4. **Archive Outdated Records:**
    - Periodically move completed queue records older than 6–12 months to a separate archive table.
    - Ensure archived data remains retrievable for reporting purposes.
5. **Backup Verification:**
    - Perform sample restores to confirm backup reliability.


Server: 192.168.3.5:3306 » Database: queue
Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers Tracking Designer Central columns
Filters
Containing the word:
Table Action Rows Type Collation Size Overhead
departments Browse Structure Search Insert Empty Drop 17 InnoDB utf8mb4_general_ci 16.0 KiB -
documents Browse Structure Search Insert Empty Drop 25 InnoDB utf8mb4_general_ci 16.0 KiB -
password_reset_codes Browse Structure Search Insert Empty Drop 1 InnoDB utf8mb4_general_ci 32.0 KiB -
requests Browse Structure Search Insert Empty Drop 37 InnoDB utf8mb4_general_ci 32.0 KiB -
staff_departments Browse Structure Search Insert Empty Drop 3 InnoDB utf8mb4_general_ci 48.0 KiB -
strands Browse Structure Search Insert Empty Drop 4 InnoDB utf8mb4_general_ci 16.0 KiB -
student_database Browse Structure Search Insert Empty Drop 45 InnoDB utf8mb4_general_ci 32.0 KiB -
users Browse Structure Search Insert Empty Drop 61 InnoDB utf8mb4_general_ci 48.0 KiB -
8 tables Sum 193 InnoDB utf8mb4_general_ci 240.0 KiB 0 B
Check all With selected:
Print Data dictionary
Create new table
Table name
Number of columns
4
Create


Server: 192.168.3.5:3306 » Database: queue » Table: departments
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Showing rows 0 - 16 (17 total, Query took 0.0006 seconds.)
SELECT * FROM `departments`
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all Number of rows: 25 Filter rows: Search this table Sort by key: None
Extra options
id name
1 Bachelor of Science in Nursing
2 Bachelor of Science in Pharmacy
3 Bachelor of Science in Physical Therapy
4 Bachelor of Science in Medical Laboratory Science ...
5 Bachelor of Science in International Hospitality M...
6 Bachelor of Elementary Education
7 Bachelor of Special Needs Education Major in Early...
8 Bachelor of Science in Criminology
9 Bachelor of Science in International Tourism Manag...
10 Bachelor of Science in Information Technology
11 Bachelor of Science in Accountancy
12 Bachelor of Science in Accounting Information Syst...
13 Bachelor of Science in Business Administration Maj...
14 Bachelor of Science in Business Administration Maj...
15 Bachelor of Science in Business Administration Maj...
16 Bachelor of Science in Psychology
17 Masters of Information Technology
Check all With selected: Edit Copy Delete Export
Query results operations



Server: 192.168.3.5:3306 » Database: queue » Table: documents

Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers

```
SELECT * FROM `documents`
```

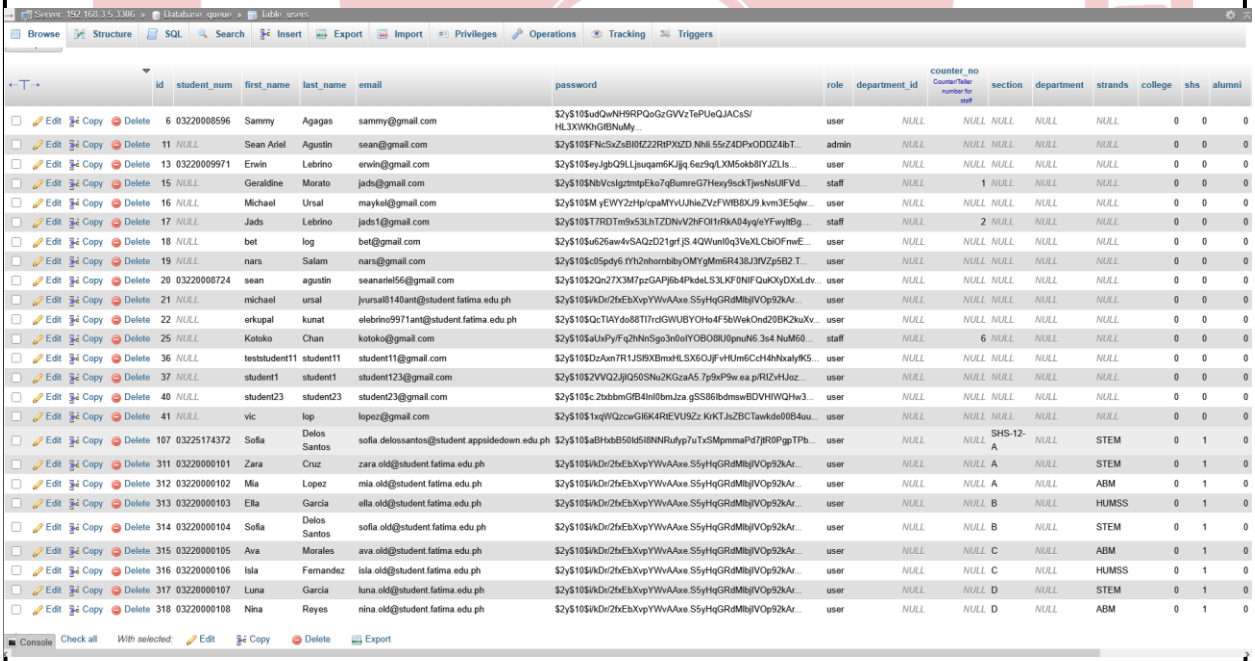
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	id	name	processing_days	fee	extra_info
Edit Copy Delete	1	Good Moral	1	0.00	NULL
Edit Copy Delete	3	Form 137	1	0.00	NULL
Edit Copy Delete	4	Transcript of Records	1	0.00	NULL
Edit Copy Delete	7	Certificate of Graduation	7	0.00	NULL
Edit Copy Delete	8	Certificate of Enrollment	1	0.00	NULL
Edit Copy Delete	9	Honorable Dismissal	3	0.00	NULL
Edit Copy Delete	10	Student Clearance	3	0.00	NULL
Edit Copy Delete	11	Certification of Units Earned	1	0.00	NULL
Edit Copy Delete	12	Certification of General Weighted Average	1	0.00	NULL
Edit Copy Delete	14	Certificate of Registration (COR)	1	0.00	NULL
Edit Copy Delete	15	Certificate of Enrollment Verification	1	0.00	NULL
Edit Copy Delete	16	Certificate of Transfer Credentials	3	100.00	NULL
Edit Copy Delete	17	Certificate of Academic Standing	2	50.00	NULL
Edit Copy Delete	18	Certificate of No Pending Case	1	50.00	NULL
Edit Copy Delete	19	Certificate of Authentication and Verification (CA...	5	200.00	NULL
Edit Copy Delete	20	Certificate of English as Medium of Instruction (E...	2	100.00	NULL
Edit Copy Delete	21	Certificate of Graduation (for Employment)	3	100.00	NULL
Edit Copy Delete	22	Certificate of Non-Issuance of Diploma	2	50.00	NULL
Edit Copy Delete	23	Certificate of School Records	2	100.00	NULL
Edit Copy Delete	24	Certificate of Attendance	1	0.00	NULL
Edit Copy Delete	25	Certificate of Admission	1	0.00	NULL
Edit Copy Delete	26	Certificate of Academic Rank (for honor students)	3	50.00	NULL
Edit Copy Delete	27	Certificate of Enrollment Status	1	0.00	NULL
Edit Copy Delete	28	Diploma (Re-Issuance)	7	500.00	NULL
Edit Copy Delete	29	CTC / Certified True Copy of Documents	2	100.00	NULL

Console

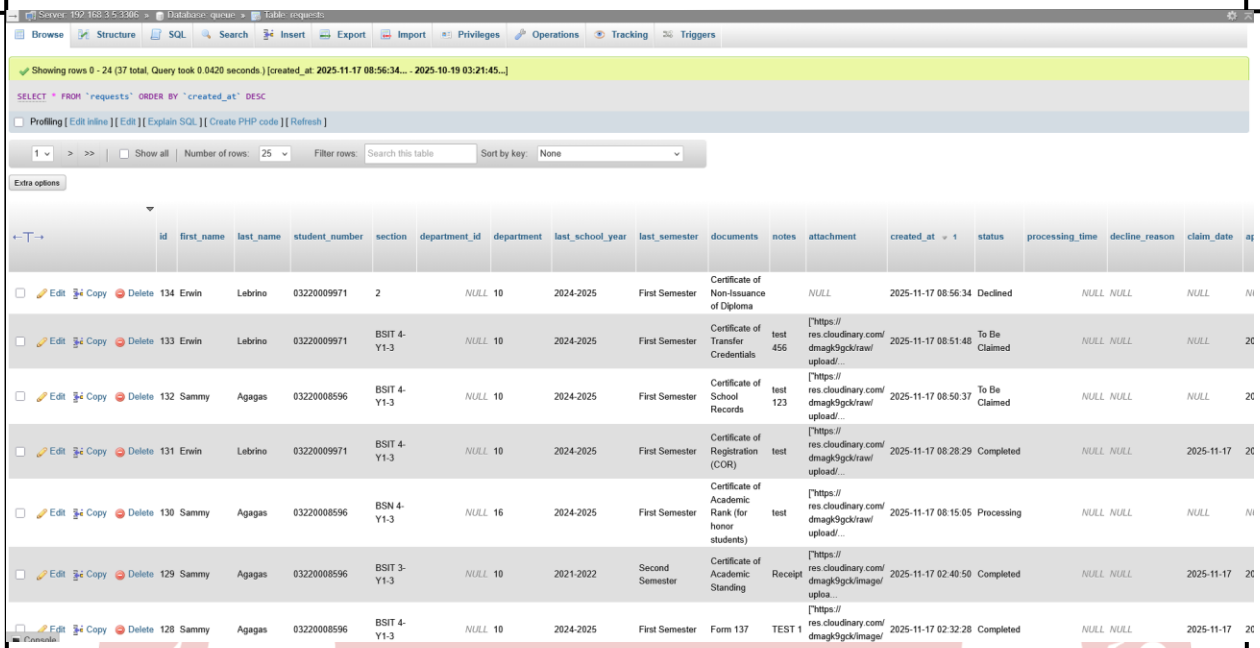


Server: 192.168.3.5:3306 » Database: queue » Table: users

Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers

	id	student_num	first_name	last_name	email	password	role	department_id	counter_no (Counter/Teller number for staff)	section	department	strands	college	shs	alumni
Edit Copy Delete	6	03220008596	Sammy	Agagas	sammy@gmail.com	$2y$10$udQwNH9RPQoGzGVVzTePUeQJACsS/HL3XWKhGfBNuMy...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	11	NULL	Sean Ariel	Agustin	sean@gmail.com	$2y$10$FNcSxZsBI0fZ22RtPXtZD.Nhli.55rZ4DPxODDZ4ibT...	admin	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	13	03220009971	Erwin	Lebrino	erwin@gmail.com	$2y$10$ey.JgbQ9LLjsuqam6KJjlq.6ez9q/LXM5okb8IYJZLls...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	15	NULL	Geraldine	Morato	jads@gmail.com	$2y$10$NbVcsIgztmtpEko7qBumreG7Hexy9sckTjwsNsUlFVd...	staff	NULL	1	NULL	NULL	NULL	0	0	0
Edit Copy Delete	16	NULL	Michael	Ursal	maykel@gmail.com	$2y$10$M.yEWY2zHp/cpaMYvUJhieZVzFWfB8XJ9.kvm3E5qlw...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	17	NULL	Jads	Lebrino	jads1@gmail.com	$2y$10$T7RDTm9x53LhTZDNvV2hFOI1rRkA04yq/eYFwyltBg...	staff	NULL	2	NULL	NULL	NULL	0	0	0
Edit Copy Delete	18	NULL	bet	log	bet@gmail.com	$2y$10$u626aw4vSAQzD21grfjS.4QWunl0q3VeXLCbiOFnwE...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	19	NULL	nars	Salam	nars@gmail.com	$2y$10$c05pdy6.tYh2nhornbibyOMYgMm6R438J3tVZp5B2.T...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	20	03220008724	sean	agustin	seanariel56@gmail.com	$2y$10$2Qn27X3M7pzGAPj6b4PkdeLS3LKF0NlFQuKXyDXxLdv...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	21	NULL	michael	ursal	jvursal8140ant@student.fatima.edu.ph	$2y$10$ikDr/2fxEbXvpYWvAAxe.S5yHqGRdMlbjlVOp92kAr...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	22	NULL	erkupal	kunat	elebrino9971ant@student.fatima.edu.ph	$2y$10$QcTlAYdo88Tl7rclGWUBYOHo4F5bWekOnd20BK2kuXv...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	25	NULL	Kotoko	Chan	kotoko@gmail.com	$2y$10$aUxPy/Fq2hNnSgo3n0olYOBO8lU0pnuN6.3s4.NuM60...	staff	NULL	6	NULL	NULL	NULL	0	0	0
Edit Copy Delete	36	NULL	teststudent11	student11	student11@gmail.com	$2y$10$DzAxn7R1JSf5XBmxHLSX6OJjFvHUm6CcH4hNxalyfK5...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	37	NULL	student1	student1	student123@gmail.com	$2y$10$2VVQ2JjlQ50SNu2KGzaA5.7p9xP9w.ea.p/RlZvHJoz...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	40	NULL	student23	student23	student23@gmail.com	$2y$10$c.2bbbmGfB4lnl0bmJza.gSS86lbdmswBDVHlWQHw3...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	41	NULL	vic	lop	lopez@gmail.com	$2y$10$1xqWQzcwGl6K4RtEVU9Zz.KrKTJsZBCTawkde00B4uu...	user	NULL	NULL	NULL	NULL	NULL	0	0	0
Edit Copy Delete	107	03225174372	Sofia	Delos Santos	sofia.delossantos@student.appsidedown.edu.ph	$2y$10$aBHxbB50ld5l8NNRufyp7uTxSMpmmaPd7jtR0PgpTPb...	user	NULL	NULL	SHS-12-A	NULL	STEM	0	1	0
Edit Copy Delete	311	03220000101	Zara	Cruz	zara.old@student.fatima.edu.ph	$2y$10$ikDr/2fxEbXvpYWvAAxe.S5yHqGRdMlbjlVOp92kAr...	user	NULL	NULL	A	NULL	STEM	0	1	0
Edit Copy Delete	312	03220000102	Mia	Lopez	mia.old@student.fatima.edu.ph	$2y$10$ikDr/2fxEbXvpYWvAAxe.S5yHqGRdMlbjlVOp92kAr...	user	NULL	NULL	A	NULL	ABM	0	1	0
Edit Copy Delete	313	03220000103	Ella	Garcia	ella.old@student.fatima.edu.ph	$2y$10$ikDr/2fxEbXvpYWvAAxe.S5yHqGRdMlbjlVOp92kAr...	user	NULL	NULL	B	NULL	HUMSS	0	1	0
Edit Copy Delete	314	03220000104	Sofia	Delos Santos	sofia.old@student.fatima.edu.ph	$2y$10$ikDr/2fxEbXvpYWvAAxe.S5yHqGRdMlbjlVOp92kAr...	user	NULL	NULL	B	NULL	STEM	0	1	0
Edit Copy Delete	315	03220000105	Ava	Morales	ava.old@student.fatima.edu.ph	$2y$10$ikDr/2fxEbXvpYWvAAxe.S5yHqGRdMlbjlVOp92kAr...	user	NULL	NULL	C	NULL	ABM	0	1	0
Edit Copy Delete	316	03220000106	Isla	Fernandez	isla.old@student.fatima.edu.ph	$2y$10$ikDr/2fxEbXvpYWvAAxe.S5yHqGRdMlbjlVOp92kAr...	user	NULL	NULL	C	NULL	HUMSS	0	1	0
Edit Copy Delete	317	03220000107	Luna	Garcia	luna.old@student.fatima.edu.ph	$2y$10$ikDr/2fxEbXvpYWvAAxe.S5yHqGRdMlbjlVOp92kAr...	user	NULL	NULL	D	NULL	STEM	0	1	0
Edit Copy Delete	318	03220000108	Nina	Reyes	nina.old@student.fatima.edu.ph	$2y$10$ikDr/2fxEbXvpYWvAAxe.S5yHqGRdMlbjlVOp92kAr...	user	NULL	NULL	D	NULL	ABM	0	1	0

Console | Check all | With selected: Edit Copy Delete Export



Server: 192.168.3.5:3306 » Database: queue » Table: requests

Browse | Structure | SQL | Search | Insert | Export | Import | Privileges | Operations | Tracking | Triggers

Showing rows 0 - 24 (37 total, Query took 0.0420 seconds.) [created_at: 2025-11-17 08:56:34... - 2025-10-19 03:21:45...]

```
SELECT * FROM `requests` ORDER BY `created_at` DESC
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

1 > >> | Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

←T→	id	first_name	last_name	student_number	section	department_id	department	last_school_year	last_semester	documents	notes	attachment	created_at ▼ 1	status	processing_time	decline_reason	claim_date	ap
Edit Copy Delete	134	Erwin	Lebrino	03220009971	2	NULL	10	2024-2025	First Semester	Certificate of Non-Issuance of Diploma		NULL	2025-11-17 08:56:34	Declined	NULL	NULL	NULL	
Edit Copy Delete	133	Erwin	Lebrino	03220009971	BSIT 4-Y1-3	NULL	10	2024-2025	First Semester	Certificate of Transfer Credentials	test 456	["https://res.cloudinary.com/dmagk9gck/raw/upload/...	2025-11-17 08:51:48	To Be Claimed	NULL	NULL	NULL	20
Edit Copy Delete	132	Sammy	Agagas	03220008596	BSIT 4-Y1-3	NULL	10	2024-2025	First Semester	Certificate of School Records	test 123	["https://res.cloudinary.com/dmagk9gck/raw/upload/...	2025-11-17 08:50:37	To Be Claimed	NULL	NULL	NULL	20
Edit Copy Delete	131	Erwin	Lebrino	03220009971	BSIT 4-Y1-3	NULL	10	2024-2025	First Semester	Certificate of Registration (COR)	test	["https://res.cloudinary.com/dmagk9gck/raw/upload/...	2025-11-17 08:28:29	Completed	NULL	NULL	2025-11-17	20
Edit Copy Delete	130	Sammy	Agagas	03220008596	BSN 4-Y1-3	NULL	16	2024-2025	First Semester	Certificate of Academic Rank (for honor students)	test	["https://res.cloudinary.com/dmagk9gck/raw/upload/...	2025-11-17 08:15:05	Processing	NULL	NULL	NULL	
Edit Copy Delete	129	Sammy	Agagas	03220008596	BSIT 3-Y1-3	NULL	10	2021-2022	Second Semester	Certificate of Academic Standing	Receipt	["https://res.cloudinary.com/dmagk9gck/image/uploa...	2025-11-17 02:40:50	Completed	NULL	NULL	2025-11-17	20
Edit Copy Delete	128	Sammy	Agagas	03220008596	BSIT 4-Y1-3	NULL	10	2024-2025	First Semester	Form 137	TEST 1	["https://res.cloudinary.com/dmagk9gck/image/	2025-11-17 02:32:28	Completed	NULL	NULL	2025-11-17	20

Console

2.2 System Backup & Recovery Process

Daily incremental backups and weekly full backups ensure minimal data loss. Recovery procedures include restoring both database and system files in case of server failure. Backup copies should be stored in secure, offsite locations or cloud storage.

To prevent data loss and downtime, a formal backup and recovery strategy must be followed.

Backup Procedures:

- **Daily Incremental Backups:** Only changes since the previous backup are stored to minimize storage usage.
- **Weekly Full Backups:** Complete backup of all system files and database tables.
- **Offsite Storage:** Store backups in a secure cloud platform or an external drive at a separate location.

Recovery Procedures:

1. Verify the integrity of backup files.
2. Restore database tables and configuration files to a staging environment first.
3. Test queue operations post-recovery before making the system live.

2.3 Security Maintenance Procedures

Security measures include monitoring for unauthorized access, enforcing strong passwords for staff accounts, removing inactive accounts, and securing hosting-level firewalls. Periodic audits prevent data breaches and protect sensitive student information. **Security Tasks Include:**

1. **Access Monitoring:** Review login attempts and detect anomalies.
2. **Account Management:** Remove inactive staff accounts and enforce strong passwords.
3. **Server-Level Security:** Maintain firewalls and verify server security settings.
4. **Data Protection:** Ensure encrypted connections via HTTPS and secure storage of sensitive data.
5. **Periodic Audits:** Conduct monthly security checks to identify vulnerabilities.

2.4 Source Code Maintenance

Source code should be maintained using **version control systems** such as Git. Developers test updates in a staging environment before deployment. Proper documentation of code changes and adherence to coding standards ensures maintainability.

Steps for Source Code Maintenance:

1. Use a **version control system** (e.g., Git) to track all changes.
2. Document changes with descriptive commit messages.
3. Test all updates in a **staging environment** before deploying to production.
4. Avoid hardcoded credentials or unencrypted sensitive information.
5. Conduct regular code reviews to identify potential bugs or inefficiencies.

2.5 Software Update & Version Control Procedures

System updates follow a controlled workflow: pull latest repository version → test in staging → deploy to production. Version control allows rollback if a deployed update introduces errors. Updates must follow a controlled workflow to avoid system downtime:

Update Workflow:

1. Pull the latest repository version to a staging environment.
2. Test all functionalities, including queue reservations and dashboard operations.
3. Deploy updates to production after approval.
4. Use version numbering to track changes and allow rollback if needed.

Note: Detailed release notes should be maintained for each update, explaining changes, bug fixes, and affected modules.

2.6 Web Application Maintenance Procedures

Maintenance includes monitoring forms for proper validation, ensuring that service types are correctly listed, checking the real-time queue updates, and validating responsiveness across different browsers. Web maintenance ensures the frontend functions as intended for students and staff.

Maintenance Steps:

1. **Form Validation:** Confirm all input forms are functioning and properly validated.
2. **Service Listings:** Check that all service types (Document Request, Clearance, Records) are accurately displayed.
3. **Queue Updates:** Verify real-time updates are reflecting correctly on student dashboards.
4. **Browser Testing:** Test the web application across multiple browsers (Chrome, Firefox, Edge) for compatibility.
5. **Interface Review:** Check responsiveness, layout, and interactive elements.

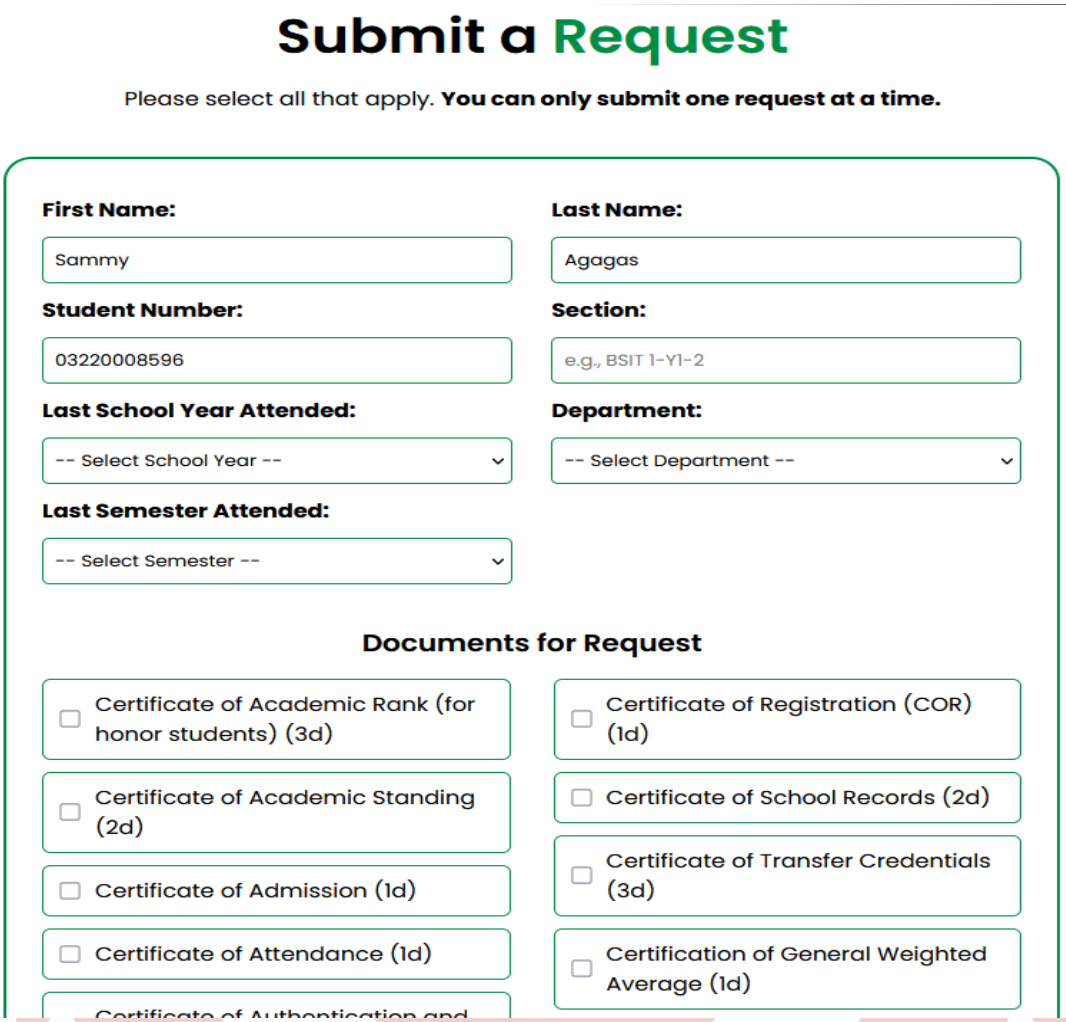


Figure 1.2 Request form

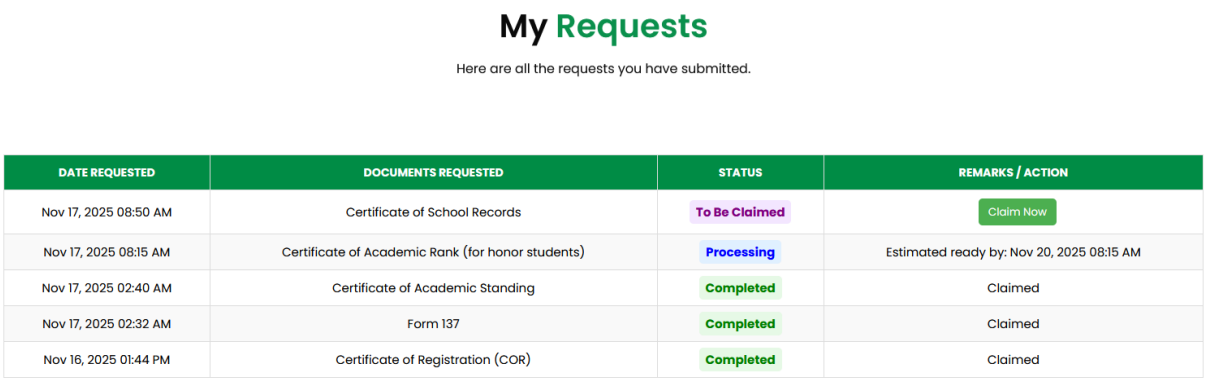
2.7 Queue Management Logic Maintenance

Ensure queues increment correctly, service types match requests, and status updates reflect actual position. Include simulation examples for peak hours. Queue logic ensures fairness and accuracy in service delivery.

Maintenance Steps:

1. Verify that **queue numbers increment correctly** as new students reserve slots.

2. Confirm **service categorization** matches student requests.
3. Ensure **status updates** reflect the correct queue position (Pending, Serving, Completed).
4. Test edge cases, such as simultaneous queue reservations or system refresh scenarios.



2.8 Queue Management Logic Maintenance

Queue numbers must increment correctly, service categorization must match requests, and status updates must accurately reflect the student's position. Staff should verify that queues are processed in the correct order.

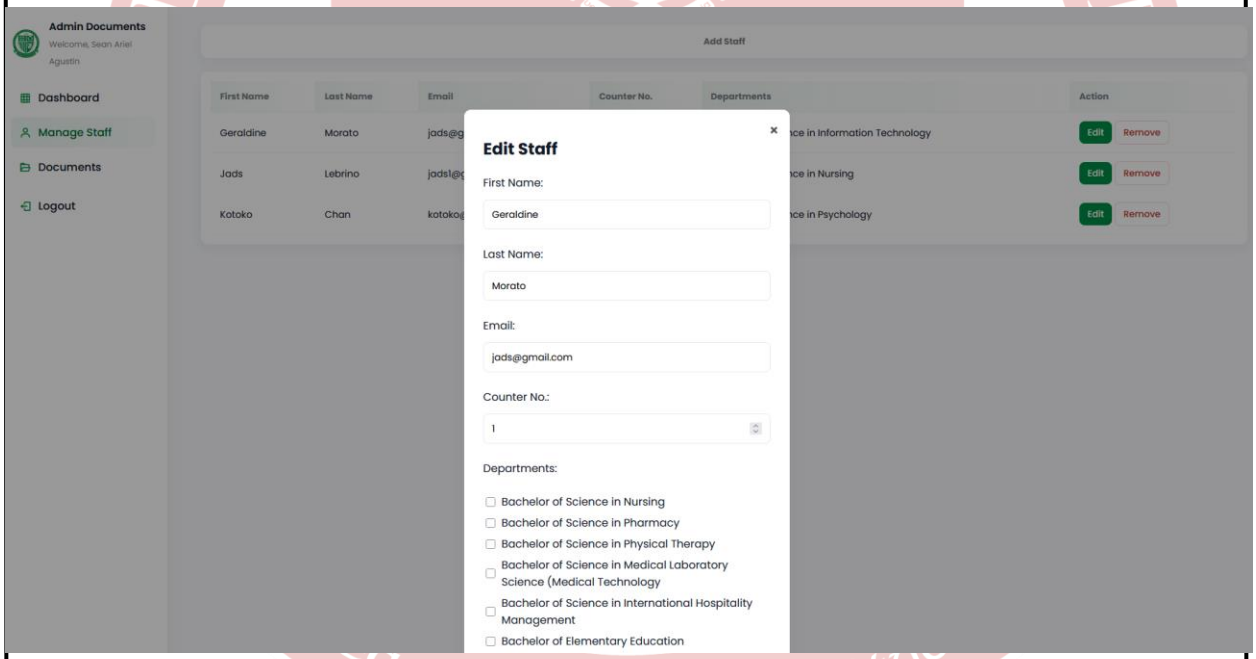
2.9 User Account Management Maintenance

Staff accounts must be properly managed, including creation, deactivation, and password resets. Login attempts should be monitored to prevent unauthorized access. Managing staff accounts is critical for security and operational efficiency.

Maintenance Steps:

1. Create accounts for new staff with proper roles and permissions.
2. Deactivate accounts for staff who have left the office.
3. Reset passwords for staff when requested.
4. Monitor login attempts for anomalies and potential security breaches.

Tip: Maintain an access control matrix to ensure users only have access to necessary functions.

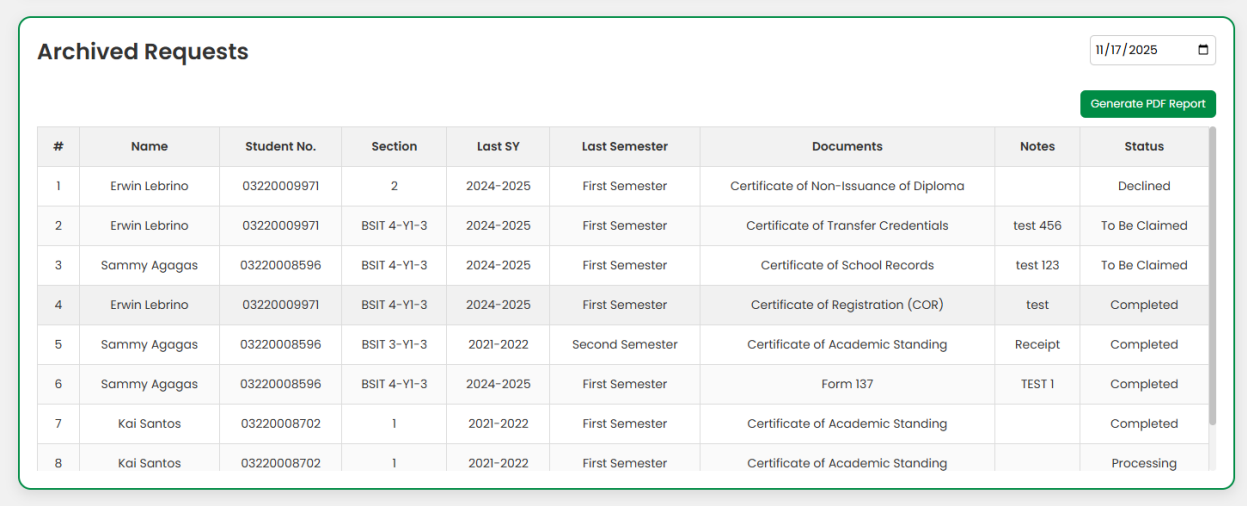


2.10 System Logs & Audit Trail Maintenance

All system actions, including queue creation and updates, are logged. Regular review of logs can detect errors, security issues, or abnormal activity. Old logs should be archived periodically. Logs track system activities and are essential for auditing and troubleshooting.

Maintenance Steps:

1. Review system logs for errors, unusual activity, and failed transactions.
2. Archive older logs periodically to reduce storage usage.
3. Document recurring errors and steps taken to resolve them.
4. Use audit trails to trace issues during troubleshooting.



Archived Requests

11/17/2025

Generate PDF Report

#	Name	Student No.	Section	Last SY	Last Semester	Documents	Notes	Status
1	Erwin Lebrino	03220009971	2	2024-2025	First Semester	Certificate of Non-Issuance of Diploma		Declined
2	Erwin Lebrino	03220009971	BSIT 4-Y1-3	2024-2025	First Semester	Certificate of Transfer Credentials	test 456	To Be Claimed
3	Sammy Agagas	03220008596	BSIT 4-Y1-3	2024-2025	First Semester	Certificate of School Records	test 123	To Be Claimed
4	Erwin Lebrino	03220009971	BSIT 4-Y1-3	2024-2025	First Semester	Certificate of Registration (COR)	test	Completed
5	Sammy Agagas	03220008596	BSIT 3-Y1-3	2021-2022	Second Semester	Certificate of Academic Standing	Receipt	Completed
6	Sammy Agagas	03220008596	BSIT 4-Y1-3	2024-2025	First Semester	Form 137	TEST 1	Completed
7	Kai Santos	03220008702	1	2021-2022	First Semester	Certificate of Academic Standing		Completed
8	Kai Santos	03220008702	1	2021-2022	First Semester	Certificate of Academic Standing		Processing

2.11 Performance Monitoring Procedures

Monitoring includes checking response times, submission success rates, and server load during peak hours. Performance tests ensure the system can handle increased traffic without lag or downtime. Monitoring system performance ensures smooth operation under all conditions.

Steps:

1. Measure page load times and server response times.
2. Monitor queue submission success rates and errors.
3. Conduct stress tests during peak hours (e.g., enrollment periods).
4. Identify performance bottlenecks and apply optimizations.

Chapter 3: Troubleshooting & Incident Response

3.1 Troubleshooting Guide

This section provides a structured approach to resolving common issues encountered by students and staff while using the Queueing Management System. Each scenario includes possible causes, step-by-step resolutions, and preventive recommendations.

Scenario 1: Queue Submission Failure

Problem: A student attempts to reserve a queue slot but receives an error or the queue number is not generated.

Possible Causes:

- Database connectivity issue.
- Server-side script error.
- Network interruption on the student's end.

Step-by-Step Resolution:

1. Verify the server status using the hosting provider's dashboard.

2. Check database connectivity by running a test query:

SELECT * FROM queues WHERE status='pending';

3. Inspect web application logs for errors at the time of the submission.
4. Test the queue submission form from multiple browsers and devices.
5. Apply fixes to backend scripts if necessary and re-test queue reservation.

Preventive Measures:

- Regularly monitor server health and database connections.
- Conduct daily tests of queue submissions.
- Apply code updates in a staging environment before production deployment.

Scenario 2: Admin Dashboard Not Loading

Problem: Staff cannot access or load the administrative dashboard.

Possible Causes:

- Server downtime or maintenance.
- Browser caching issues.
- Incorrect staff account permissions.

Step-by-Step Resolution:

1. Verify server uptime and confirm there are no scheduled downtimes.

2. Clear the browser cache and cookies, or test in an alternative browser.

3. Check staff account roles and permissions in the database.

4. Inspect server error logs for indications of script failure or database timeouts.

5. Restart server services if necessary and verify dashboard accessibility.

Preventive Measures:

- Maintain a maintenance log for server updates and scheduled downtime.
- Conduct periodic user permission audits.
- Optimize server scripts to prevent slow response times.

Scenario 3: Incorrect Queue Number Increment

Problem: Queue numbers are skipped, duplicated, or out of order.

Possible Causes:

- Database transaction conflicts.
- Backend script logic errors.
- Concurrent queue submissions causing race conditions.

Step-by-Step Resolution:

1. Inspect queue logic scripts to ensure proper increment handling.

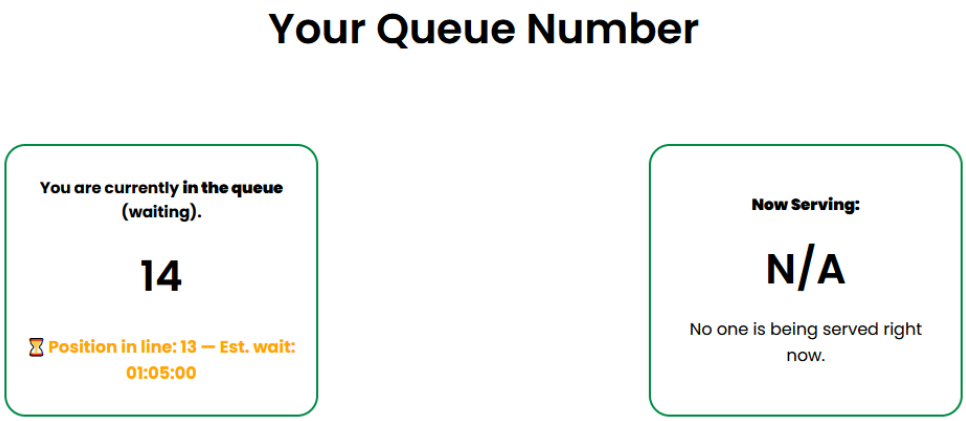
2. Verify database primary key integrity for the queue table.

3. Test multiple simultaneous queue submissions in a staging environment.

4. Apply fixes to scripts to handle concurrency safely.

Preventive Measures:

- Schedule weekly testing of queue logic.
- Implement database constraints to enforce uniqueness of queue numbers.



Scenario 4: Pending Requests Not Updating on Dashboard

Problem: Queue statuses remain static and do not reflect completed or ongoing services.

Possible Causes:

- Backend queue update scripts not executing properly.
- Database triggers or cron jobs failing.
- Network connectivity issues between server and database.

Step-by-Step Resolution:

1. Verify the execution of scheduled update scripts.
2. Check database for pending updates or locks that prevent row changes.
3. Confirm server connectivity and network stability.
4. Test status updates manually in a staging environment.
5. Apply fixes to backend scripts and re-run updates.

Preventive Measures:

- Schedule automated status update checks.
- Log all queue status changes for auditing and debugging.

3.2 Error Codes & Incident Response

Define error codes for server, database, and notification issues. Establish procedures to quickly restore functionality and notify staff of ongoing problems.

Error Code	Description	Recommended Action
DB01	Database connection failed	Check server status, verify credentials, restart database service
WEB01	Frontend form not submitting	Inspect form validation, browser compatibility, backend script

Error Code	Description	Recommended Action
QUEUE01	Queue number not generated	Verify queue increment logic, test concurrent submissions
PERF01	Slow response time	Check server load, optimize queries, verify network speed

Incident Response Steps:

1. Identify the error using logs or error codes.
2. Assign responsibility to the relevant team (admin, developer, registrar staff).
3. Resolve the issue in a staging environment if possible.
4. Deploy fixes to production.
5. Document the incident, root cause, resolution steps, and preventive measures.

3.3 Preventive Maintenance Schedule

Preventive maintenance aims to reduce the likelihood of errors occurring. Activities include:

- Weekly checks of queue submissions and dashboard functionality.
- Monitoring server performance, CPU load, and memory usage.
- Validating database integrity and queue number sequencing.
- Periodic review of user accounts and permissions.

3.4 Corrective Maintenance Procedures

Corrective maintenance addresses detected issues promptly to minimize downtime.

Steps:

1. Identify and isolate the problem through logs and error reports.
2. Apply fixes in a controlled staging environment.
3. Test fixes thoroughly before deploying to production.
4. Document corrective actions and update maintenance logs.

Example: If a queue fails to update, staff would:

- Check the queue logic scripts.
- Apply the correction in staging.
- Verify that pending queues now update correctly.
- Deploy the fix to production.

3.5 Testing Procedures After Maintenance

After maintenance or corrective actions, verify system integrity through structured testing:

1. **Queue Submission Test:** Submit multiple test requests and verify queue numbers increment correctly.
2. **Dashboard Test:** Check that pending and completed queues display properly.
3. **Database Verification:** Confirm records are stored accurately with correct timestamps.

4. **Access Control Test:** Ensure staff accounts function with appropriate roles and permissions.

Chapter 4: Documentation & Compliance

4.1 Documentation Maintenance

Maintaining comprehensive documentation is critical for ensuring system sustainability, knowledge transfer, and operational continuity. Documentation should reflect the current state of the system, including updates to the web interface, backend logic, and administrative processes.

Key Documentation Activities:

1. **Update User Manuals and Guides:**
   - Include updated screenshots for all interfaces (login pages, queue reservation, dashboard).
   - Reflect any changes in service types, queue logic, or dashboard functionalities.
2. **Track System Changes:**
   - Record all updates made to the backend scripts, database schema, or frontend design.
   - Maintain detailed version histories to support troubleshooting and rollback if necessary.
3. **Maintain Procedure Logs:**

- Document routine maintenance activities, such as database checks, server monitoring, and system testing.
- Include procedural steps with references to relevant diagrams or figures.

4. **Create Flow Diagrams and Charts:**
   - Visual representations of queue processing logic, user workflows, and system architecture help new staff understand system operations quickly.

Benefit: Regular documentation updates reduce errors, improve onboarding of new staff, and facilitate audits.

4.2 Data Privacy & Compliance Maintenance

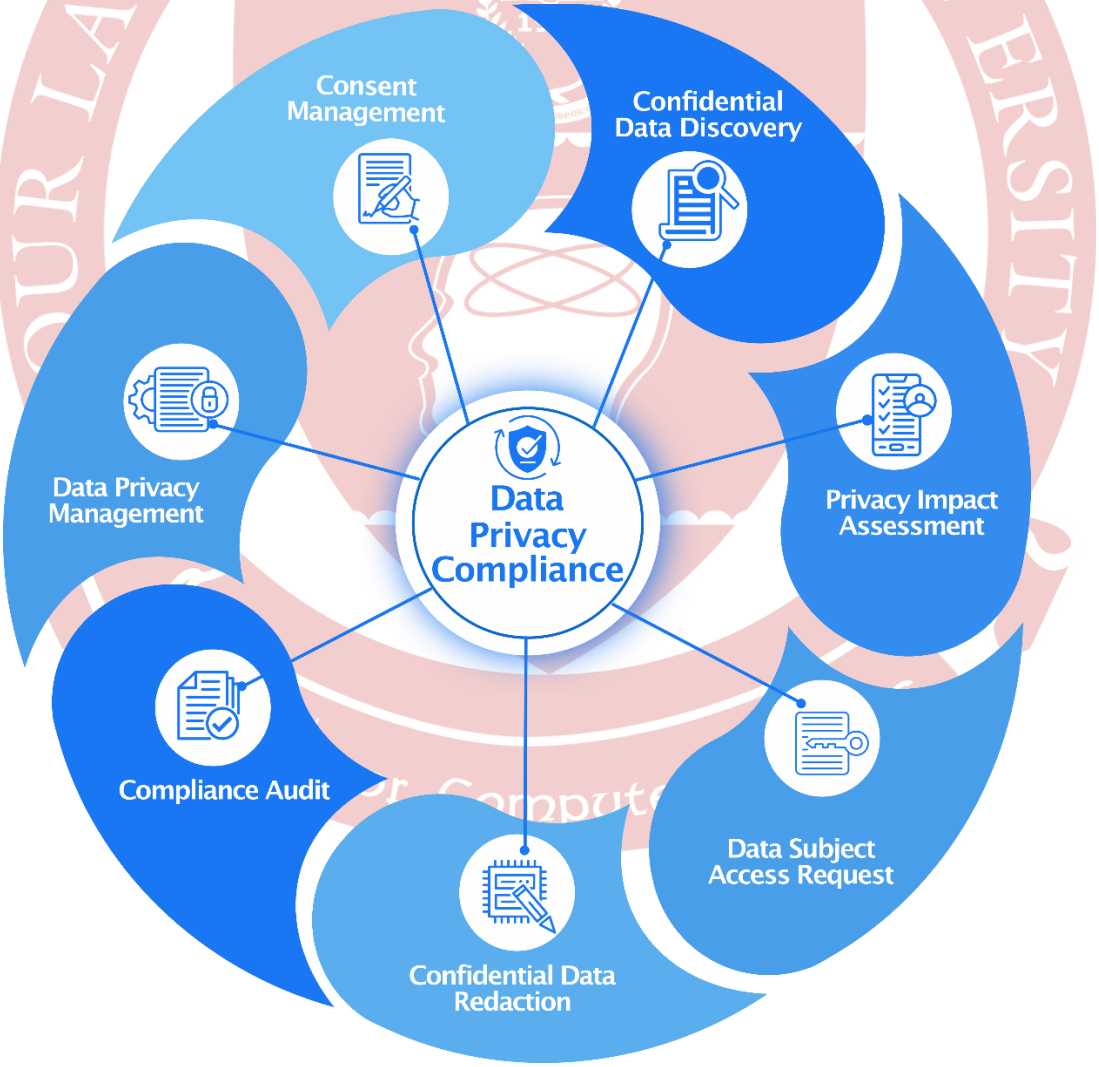
Compliance with data privacy laws is essential because the system handles sensitive student information.

Compliance Requirements:

- **Philippine Data Privacy Act (RA 10173):** Protect student data through secure storage, encrypted connections, and controlled access.
- **Access Control:** Ensure staff accounts are assigned roles appropriate to their responsibilities.
- **Data Retention:** Archive completed queue records as per institutional policy and delete outdated information securely.
- **Encryption:** All personal data stored in the database or transmitted via the web must use encryption standards such as HTTPS and hashed passwords.

Maintenance Tasks:

1. Conduct monthly audits of student data storage and access logs.
2. Confirm that no unauthorized user accounts exist.
3. Validate that data retention policies are correctly implemented in the database.
4. Review encryption protocols and update as necessary.



Outcome: Ensures that the system protects student information and meets national and institutional compliance standards.

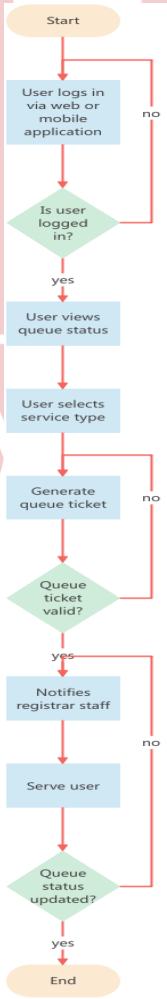
4.3 System Expansion & Scalability Plan

Planning for future growth ensures that the system remains functional as student numbers increase or additional services are added.

Scalability Considerations:

1. **Database Scalability:**
   - Ensure the database can handle increased queue volumes and more service categories.
   - Use indexing, query optimization, and database partitioning to maintain performance.
2. **Server and Hosting Scalability:**
   - Monitor server CPU, RAM, and storage usage.
   - Consider cloud-based hosting or upgrading server specifications to handle increased load.
3. **Queue Logic Expansion:**
   - Add support for additional queue types, such as scholarship processing or certificate issuance.
   - Ensure logic adjustments do not disrupt existing queue management.

4. **Frontend Scalability:**
    - Maintain responsive design for new service types or additional dashboard features.
    - Test across multiple browsers to ensure consistent user experience.
5. **Documentation & Training:**
    - Update all guides, diagrams, and workflows to reflect changes in services or system architecture.
    - Conduct training sessions for staff to ensure smooth adoption of new features.
    -



4.4 Final Recommendations

This section provides guidance to ensure long-term system reliability and operational excellence. Maintain regular audits, prompt issue reporting, and continuous updates to documentation for long-term system reliability.

Recommendations Include:

1. **Regular System Audits:**
    - Perform weekly checks of core functionalities, including queue processing, database integrity, and administrative access.
2. **Continuous Documentation Updates:**
    - Ensure every software or logic update is reflected in manuals, flowcharts, and screenshots.
3. **Staff Training and Knowledge Sharing:**
    - Conduct periodic refresher sessions for staff to ensure they are familiar with updates and troubleshooting procedures.
4. **Feedback Loop Implementation:**
    - Collect feedback from students and staff to identify areas for improvement in the queueing process.
    - Document feedback in logs and plan corrective actions accordingly.

5. **Future-Proofing:**
    - Plan for emerging needs, such as additional services or integration with other university systems.
    - Evaluate new technologies that may improve performance or user experience.

Outcome: Following these recommendations ensures that the system remains reliable, efficient, and adaptable over time.

Chapter 5: Supporting Information

5.1 Glossary of Technical Terms

Term	Definition
Queue Number	**A unique identifier assigned to each student request for a service. This number determines the order in which students are served at the Registrar Office.**
Registrar Admin Dashboard	**A web-based interface used by administrative staff to view, manage, and process student queues. It includes functionalities such as queue status monitoring, request management, and service categorization.**
Database Backup	**A copy of the database stored in a secure location for disaster recovery purposes. Backups prevent data loss in case of system failure or corruption.**

Term	Definition
Staging Environment	**A separate test environment that mirrors the production system. Updates and fixes are applied here first to prevent potential disruptions to live operations.**
Audit Trail	**A sequential record of system activities, including queue creation, updates, and administrative actions. Audit trails help in accountability, troubleshooting, and compliance verification.**
Preventive Maintenance	**Scheduled inspections and tests performed to prevent system failures before they occur. Examples include database checks, queue functionality tests, and server health monitoring.**
Corrective Maintenance	**Actions taken to fix problems after they occur, ensuring the system returns to normal operation. For example, correcting queue logic errors or resolving dashboard malfunctions.**
Version Control	**A system used to track and manage changes to the source code. Allows developers to roll back updates, maintain multiple versions, and collaborate efficiently.**
Server Hosting	**The infrastructure where the web-based queueing system is deployed. Hosting includes storage, processing, and network capabilities needed to run the system reliably.**

Term	Definition
Data Privacy Compliance	**Adhering to legal requirements for handling personal data, ensuring it is collected, stored, and transmitted securely. Includes following the Philippine Data Privacy Act (RA 10173).**

5.2 References

A formal reference section enhances academic credibility. Include primary laws, industry best practices, and academic sources.

Sample References:

1. Philippine Data Privacy Act, Republic Act No. 10173. (2012). Official Gazette of the Republic of the Philippines.
2. Wavetec, Inc. (2020). *Best Practices in Digital Queue Management Systems*. Wavetec Blog. Retrieved from https://www.wavetec.com/blog
3. TermInApp (2021). *Ten Best Practices When Running an Online Queuing System*. Retrieved from https://www.terminapp.com/blog
4. HelpJuice Knowledge Base. (2019). *User Documentation Best Practices*. Retrieved from https://helpjuice.com/blog/user-documentation
5. Laudon, K.C., & Laudon, J.P. (2021). *Management Information Systems: Managing the Digital Firm* (17th Edition). Pearson Education.
6. Chen, H., Chiang, R.H.L., & Storey, V.C. (2012). *Business Intelligence and Analytics: From Big Data to Big Impact*. MIS Quarterly, 36(4), 1165–1188.