

UNIVERSITY OF AGRICULTURE, FAISALABAD

Department of Computer Science

SYSTEM DEVELOPMENT DOCUMENT

CampusBridge UAF

Centralized Student E-Services & Workflow Routing Platform

Modules & functions | **Database & workflows** | **Plan & testing**

PROJECT TYPE	Final Year Project (web-based information system)
SUPERVISOR	Ms. Raheela Nasim
TEAM	Muhammad Hussnain | Tehseen Qaiser | Abdullah Ghaffar
DOCUMENT	Complete design and development document: modules, database, workflows, security, plan, testing, deployment
STATUS	Version 2.0 - November 2026. Describes the implemented Student E-Services Portal v2.0

Contents

1. Introduction 2
2. System Overview 2
3. Architecture 3
4. Module Breakdown 5
5. Module Specifications 5
6. Database Design 9
7. Roles and Permissions 10
8. Key Workflows 11
9. Security Design 12
10. Development Plan 12
11. Testing Strategy 13
12. Deployment and Maintenance 13
13. Risks and Mitigation 14
14. Future Enhancements 14

1 Introduction

This document explains how CampusBridge UAF is built: what the system does, how it is organised into modules, how data is stored, how the main workflows run, how access is protected, and how testing and deployment are carried out. It is written so that a reader who has never seen the project, such as the supervisor or an examiner, can understand the complete system from this document alone.

11 Modules

13 Database tables

8 Default roles

7 Default services

1.1 Purpose

To give one reference for the complete project: scope, architecture, module design, database, security, plan and testing.

1.2 Intended readers

- Supervisor and examiners: to judge the design, scope and progress.
- Developers (the team): to extend the modules consistently.
- Future maintainers at the university: to understand and operate the system.

1.3 Terms used

Term	Meaning
Service	An e-service a student can request, such as Wi-Fi access, DMC or Student ID card.
Request / application	One student's submission for one service, with a tracking code and a status.
Desk / staff	A university employee (IT, DSA, Library, Exam, LMS) who processes requests.
Campus Admin	Administrator of one or more campuses: all services, own campuses only.
Super Admin	The administrator who manages everything: staff, roles, services, campuses, settings.
AG No.	The student registration number, for example 2023-ag-10110.
CNIC	National identity card number, format 12345-1234567-8.
OTP	One-time password: a 6-digit code emailed to verify an email address.
PWA	Progressive Web App: a website that can be installed on a phone like an app.

2 System Overview

2.1 Problem

At the University of Agriculture, Faisalabad, students request services (DMC, ID card, email account, Wi-Fi, library membership, fee instalments, LMS) with paper forms and visits to several offices. There is no way to see the status of a request, officials sort paper and verify receipts by hand, and no central record shows delays or past approvals.

2.2 Solution

CampusBridge UAF is a web platform on which students apply online, upload documents and follow the request until it is completed. Each request goes automatically to the correct desk of the correct campus. Staff review it digitally and the student is notified. Every important action is recorded in an audit log. A chatbot on the public and student pages answers common questions and tracks applications.

2.3 Objectives

- Remove physical queues and repeated office visits.
- Give students live status tracking for every request, also from the chatbot.
- Reduce manual work and errors for staff.
- Let the Super Admin add or change services without changing code or the database structure.
- Support several campuses with campus-limited staff and Campus Admins.
- Provide reports, an audit trail, backups and a secure, mobile-friendly experience.

2.4 Users and roles

Role	Who	Main abilities
Student	Enrolled UAF student	Register, apply, track, edit a returned application, download the application PDF, manage the profile, use the chatbot.
Desk staff (it_admin, dsa_admin, library_admin, exam_admin, lms_admin, custom roles)	University employees	See requests of the services their role allows and of their assigned campuses; change status within the statuses allowed to them; download attachments. dsa_admin may also manage students.
Campus Admin (campus_admin)	Campus administrator	All services, own campuses only: requests, students, reports, staff of the campus.
Super Admin (super_admin)	System owner	Everything, for all campuses: staff, roles, students, users, services, campuses, reports, site settings, backups, audit logs.

3 Architecture

The system follows a layered, MVC-style architecture on PHP and MySQL. Pages (views) call controllers, and controllers use helpers and services. Each layer talks only to the layer below it.

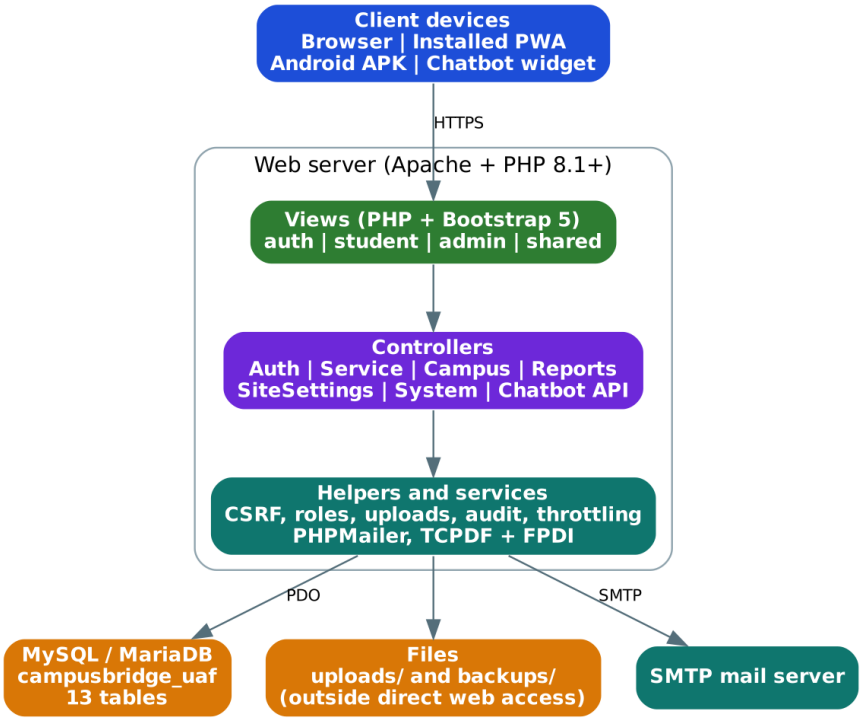

Fig. 1 - Layered architecture.

3.1 Technology stack

Layer	Technology	Why it was chosen
Front-end	HTML5, CSS3, Bootstrap 5, JavaScript, Chart.js	Responsive on desktop and phone, no heavy framework needed.
Back-end	PHP 8.1+ with controllers and helper functions	Runs on common university and shared hosting, easy to maintain.
Database	MySQL / MariaDB (InnoDB)	Relational data with foreign keys and transactions.
Libraries	PHPMailer, TCPDF + FPDI, Composer	Proven libraries for mail and PDF.
Mobile	PWA (manifest + service worker) and an Android APK wrapper	One code base for web and mobile.
Tools	Git/GitHub, phpMyAdmin, VS Code, XAMPP	Collaboration and local testing.

3.2 Folder structure

Folder / file	Content
views/auth	login, register, verify_email, verify_otp, forgot_password, reset_password
views/student	dashboard, service_list, apply_service, view_application
views/admin	dashboard, requests, review_request, manage_staff, manage_students, manage_users, manage_roles, manage_campuses, service settings, site settings, reports
views (shared)	profile, secure download, maintenance page, offline page
controllers	Auth, Service, Campus, Reports, SiteSettings, System
includes	bootstrap (headers, CSP, sessions), helpers, header / footer, PDF generator, install prompt, throttling
mailer	Mailer class built on PHPMailer
assets/js	chatbot.js (assistant widget), service worker, install prompt scripts
config, database	config/local.php (created by the installer), schema.sql, seed.sql, migrations
uploads, storage	Uploaded documents and stored backups (not directly reachable)
scripts	create_superadmin.php, reset_admin_password.php (command line)
install.php	Single-file installer; deletes itself after a successful install

4 Module Breakdown

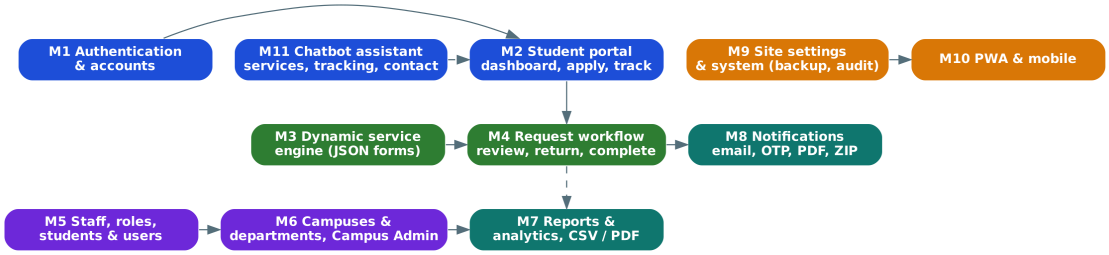

Fig. 2 - The eleven modules. Blue: student side. Green: service and workflow. Purple: administration. Teal: reports and notifications. Orange: system.

Module	Purpose	Main files
M1 Authentication & accounts	Register with CNIC, verify email, log in, lockout, reset password, forced password change	views/auth, AuthController
M2 Student portal	Dashboard with status filters, apply, track, edit returned, profile	views/student, profile
M3 Dynamic service engine	Services and form fields stored as data	service_configurations, ServiceController
M4 Request workflow	List, filter, review, return, reject, complete, downloads	requests, review_request
M5 Staff, roles, students & users	Create staff, custom roles, manage students and all users	manage_staff, manage_roles, manage_students, manage_users
M6 Campuses & departments	Campus setup, departments, Campus Admins, staff campus access	CampusController, manage_campuses
M7 Reports & analytics	Dashboards, six reports, CSV and PDF export	reports, ReportsController
M8 Notifications & documents	OTP, credentials, status and completion emails; PDF and ZIP	Mailer, PDF generator
M9 Site settings & system	Branding, banner, SMTP, maintenance, backup, audit logs, reset	site settings, SystemController
M10 PWA & mobile	Install prompt, offline page, APK	manifest, service worker
M11 Chatbot assistant	Services, how to apply, tracking, contact	index.php, chatbot.js, chatbot API

5 Module Specifications

Each module lists its purpose, main functions and the business rules it enforces.

5.1 M1 - Authentication and accounts

Function	Purpose
register()	Validates registration data (AG No., name, email, CNIC, degree, campus, department of that campus, password), checks duplicates, then sends an OTP or creates the account directly when email verification is switched off.
verifyEmailOtp()	Checks the OTP (hashed, 15-minute expiry, 5 attempts) and creates the student account.
login()	Signs in with registration number or email; blocks inactive accounts; counts failed attempts per IP and account in auth_throttle and locks sign-in for a while after repeated failures.
requestOtp() / verifyOtp() / resetPassword()	Forgot-password flow: send a code, verify it, set a new password.

Function	Purpose
updatePassword()	Change password from the profile (current + new + confirm).
must_change_password	Admin-issued temporary passwords force a new password at first login.

Rules: AG No. format 2023-ag-10110; CNIC 12345-1234567-8 and unique; password at least 8 characters with a letter and a number, not one repeated character and not a common password; passwords and OTPs stored hashed; idle timeout 30 minutes for staff and 8 hours for students; sessions end when an account is deactivated or its role or password changes.

5.2 M2 - Student portal

Function	Purpose
submitRequest()	Builds answers from the service definition, validates required fields and uploads, blocks a second open request for the same service, saves the request with the student's campus and creates the tracking code (for example CB-DMC-000023).
Edit and resubmit	A request in the Returned status can be edited and sent again.
getStudentRequests()	Lists the student's own requests with a status rail and filter chips.
Application PDF	Official form and summary generated with TCPDF + FPDI, downloadable from the details page.
Profile	Edit name, phone, CNIC, degree and department; change password.

Rules: a student sees only their own requests; uploads are images (jpg, jpeg, png, gif), at most 5 MB each.

5.3 M3 - Dynamic service engine

Each service is one row in `service_configurations`: title, enabled flag, fee, guidance, downloadable form, the list of form fields (text, number, date, select, textarea, file), the fields staff fill on completion, the completion message template and the roles allowed to process it. The Super Admin edits all of this in Service Settings, can create a new service, and the student form is generated from it. The chatbot reads the same table, so a new service appears there at once.

5.4 M4 - Request workflow and review

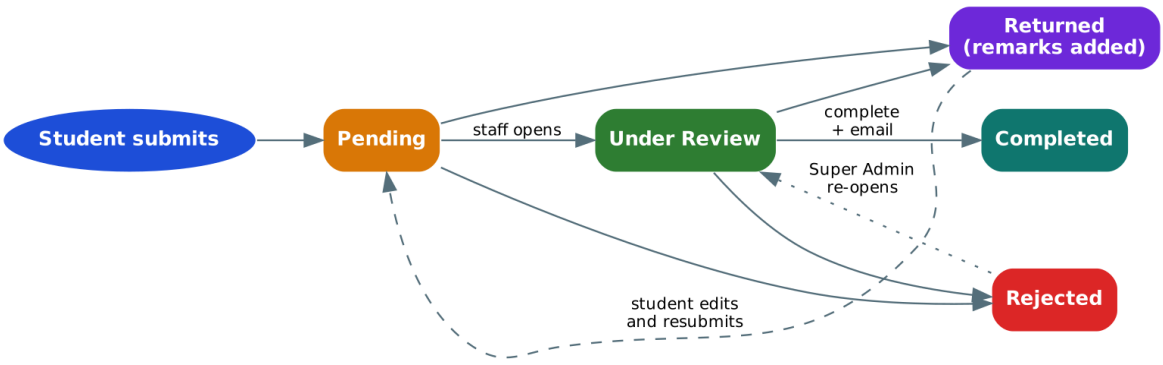

Fig. 3 - Request status lifecycle.

Rule	Behaviour
One access rule	can_access_request(): the owner student, or Super Admin, or Campus Admin of that campus, or staff whose role is allowed for the service AND whose campus matches AND whose manageable statuses allow the change.
Allowed moves	Pending to Under Review, Rejected or Returned; Under Review to Completed, Rejected or Returned; Returned to Rejected. Completed is final. Only the Super Admin can re-open a Rejected request.
Restricted staff	Staff with limited manageable statuses move a request one step at a time and cannot reject.

Rule	Behaviour
Remarks and completion	Remarks can be edited while the request is open. Completion fields and the message template come from the service settings; the student is emailed.
Lists	Search, filters for campus, service and status, status counters, pagination; back-links keep the filters.
Downloads	Secure download endpoint with the same access check; staff can download a ZIP of the application PDF and all uploaded files.

5.5 M5 - Staff, roles, students and users

Function	Purpose
Manage staff	Create (name, email, phone, CNIC, role, campuses, manageable statuses, temporary password), update, reset password, deactivate, reactivate, delete.
Manage roles	Create, update and delete custom roles; system roles are protected.
Manage students	Campus-scoped list with search, filters and bulk actions; edit the record, reset password (optionally set one), reactivate, delete. Allowed only for super_admin, campus_admin and dsa_admin.
Manage users	Super Admin overview of all accounts with the same tools.
Profile completion	Staff must complete name, email, phone and CNIC before using the panel.

5.6 M6 - Campuses and departments

Function	Purpose
Campus and department CRUD	Add, edit, disable campuses (code, address) and their departments (name, code).
Campus Admins	Add, update and remove campus admins from the Campuses page; several per campus are allowed.
Staff campus access	staff_campuses links each staff member to one or more campuses.
can_access_campus() / request filter	Applies the campus rule to every request query, so lists, counters and reports stay consistent.

Rules: a staff member with several campuses sees a campus filter; opening or downloading a request outside their campuses is refused even if the number in the address is changed.

5.7 M7 - Reports and analytics

Report	Content
Staff	Staff list by role, status and campus.
Students	Students by campus, department, degree.
Services	Requests per service with outcome counts.
Applications	Applications by status, service, campus and date.
Campuses	Requests and students per campus.
Audit	Who did what, when and from which IP (Super Admin only).

Reports have date-range presets and a campus filter, and export to CSV and PDF. Dashboards show counters per status, a status chart, a service chart and a monthly trend chart. Campus Admins see their campus only.

5.8 M8 - Notifications and documents

Item	Purpose
OTP emails	Registration and password-reset codes.
Credentials email	Login details for new staff or service accounts.
Status and completion emails	Sent when the status changes or a request is completed, from the service template (can be switched off).

Item	Purpose
Test email	Checks the SMTP settings.
PDF generator	Application PDF and report PDFs with the university header.

5.9 M9 - Site settings and system administration

Section	Content
Branding and identity	Site name, tagline, logo upload.
Contact information	Email, phone, address, website, footer note.
Email and notifications	Email verification on/off; completion email on/off.
Announcement banner	On/off, text and type (info, warning ...).
SMTP server	Host, port, encryption, user, password (stored encrypted), from address, test email.
Maintenance mode	Message and expected time; the Super Admin can still log in.
Backup and restore	Create, download, restore and delete backups; a safety backup is created before a restore or a system reset.
Audit logs	Every important action with user, time and IP; exportable.
System reset	Clears operational data after confirmation and a safety backup.

5.10 M10 - PWA and mobile

A web manifest, a service worker with an offline page and an install prompt let a student add CampusBridge to the phone's home screen. An Android APK wraps the same site. The layout is responsive, with a slide-in menu and card-style tables on phones.

5.11 M11 - Chatbot assistant

A floating assistant on the home page and student pages, written as one script (chatbot.js) and a small server endpoint. It works the same on desktop and phone, where it opens full-screen.

Feature	How it works
Welcome and quick replies	Tiles for Services, How to apply, Track status and Contact support; typing indicator, timestamps, restart and close.
Services	Lists the enabled services from the database with fee, guidance, required documents and an Apply button.
Track status	By tracking code (for example CB-WIFI-000123), or by AG No. plus CNIC. Both must belong to the same student. Shows the first name and up to 10 latest applications with a status stepper.
Privacy	Never shows remarks or personal data; the CNIC is never stored; lookups are rate limited per IP and per AG No.
Free text	Keyword understanding for common questions; falls back to the contact details.

5.12 Shared helpers

Helper	Purpose
csrf_token() / csrf_verify()	Protects every form and action against forged requests.
throttle_hit() / throttle_locked_for() / throttle_clear() / client_ip()	Login and chatbot rate limiting.
require_login() / require_role()	Access control for pages and actions.

Helper	Purpose
can_access_request()	The single access rule for one request.
handle_secure_upload ()	Validates and stores uploaded files.
generate_tracking_code() / generate_strong_password()	Codes and passwords.
log_audit()	Writes important actions to audit_logs.

6 Database Design

The database **campusbridge_uaf** has 13 tables (InnoDB, utf8mb4) and 15 foreign keys. The ER diagram and data dictionary are in the Database Design document and in 2_Database.

Table	Purpose	Key relationships
users	Students, staff, Campus Admins and Super Admin (cnic, must_change_password, manageable_statuses)	role -> roles.name; campus_id -> campuses; department_id -> departments
roles	Role list; system roles protected	referenced by users.role
campuses	Campuses (code, address)	parent of departments, users, requests
departments	Departments of one campus (code)	campus_id -> campuses
staff_campuses	Campuses each staff member manages	staff_id -> users; campus_id -> campuses
service_configurations	Services, fees, form fields, completion fields, allowed roles	matched to requests by service_type
service_requests	Applications with status, answers, files and campus	user_id -> users; campus_id -> campuses; reviewed_by -> users
credentials_dispatch	Login details created on completion	request_id -> service_requests
audit_logs	Who did what, from which IP	admin_id -> users; target_request_id -> service_requests
auth_throttle	Login and chatbot lockout counters	key: scope + key_hash
password_resets, email_verifications	Hashed OTP records	user_id -> users (resets)
site_settings	Branding, contact, banner, maintenance, email options, SMTP	updated_by -> users

Databases from earlier versions are upgraded with `update_campuses_departments.sql` (campus feature) and then `update_to_v2.sql` (all version 2.0 tables and columns). Both can be run more than once safely.

7 Roles and Permissions

Capability	Student	Desk staff	Campus Admin	Super Admin
Register, log in, edit profile	Yes	Yes	Yes	Yes
Apply and track own requests, chatbot	Yes	-	-	-
See requests	Own only	Allowed services, own campuses	All services, own campuses	All
Change request status	-	Yes (allowed statuses)	Yes	Yes
Download attachments and ZIP	Own	Own scope	Own campuses	All
Manage students	-	dsa_admin only	Own campuses	All
Create staff and assign campuses	-	-	Own campus	Yes
Manage roles, services, form fields	-	-	-	Yes
Manage campuses and departments	-	-	-	Yes
Reports	-	-	Own campuses	All, with audit
Site settings, backup, maintenance	-	-	-	Yes

8 Key Workflows



Fig. 4 - Applying for a service.

8.1 Applying for a service

- The student logs in and opens Services, then chooses a service.
- The form is built from the service's field list. The student fills it and uploads documents.
- The system validates the data, saves the request with the student's campus and creates a tracking code.
- The request appears in the queue of the staff whose role and campus match.
- Staff review it and move it to Under Review, Returned (with remarks), Rejected or Completed.
- The student receives an email and sees the new status on the dashboard or in the chatbot.

8.2 Completing a request that creates login details

For Wi-Fi, university email and LMS, staff enter the credentials in the completion fields. The system saves them in credentials_dispatch, builds the final message from the service template and emails the student.

8.3 Password recovery

The user enters an email, receives a 6-digit code (valid 15 minutes, 5 attempts), enters it, and sets a new password that follows the password policy.

8.4 Creating a staff or Campus Admin account

The Super Admin (or a Campus Admin for their own campus) enters name, email, phone, CNIC, role, campuses and the statuses the person may set. A temporary password is emailed. At first login the person must change it and complete the profile.

8.5 Tracking from the chatbot

The student opens the chatbot, chooses Track status and enters either the tracking code or AG No. and CNIC. The server checks the pair, applies rate limits and returns only the status, service and date.

9 Security Design

Threat	Protection
Forged form submissions	CSRF token on every form and action.
SQL injection	PDO prepared statements everywhere.
Cross-site scripting	Output escaped once; Content-Security-Policy and other security headers.
Password guessing	Login throttling with lockout; chatbot lookups rate limited; password policy.
Password theft	Passwords and OTPs stored hashed; SMTP password stored encrypted; temporary passwords must be changed.
Unauthorised access	Role checks on every page; one access rule for every single request and download, including campus and status checks.
Malicious uploads	Image types only, 5 MB limit, stored outside direct URLs and served through a controller.
Session theft	HttpOnly, SameSite and Secure cookies; idle timeouts; sessions invalidated on deactivation, role change or password change.
Data loss	Stored backups, safety backup before restore and system reset, restore function.
Tampering and accountability	Audit log of important actions with user, time and IP.

10 Development Plan

10.1 Methodology

The team uses an Agile, iterative approach with short cycles. Each cycle ends with a working increment that is shown to the supervisor, so requirements can be corrected early. Work is tracked in Git with one branch per module.

10.2 Phases and status

Phase	Main tasks	Deliverable	Status
1 Requirements	Study the paper process, meet offices, list requirements	Requirement list, proposal	Done
2 Design	Architecture, use cases, data flow, UI design	Design diagrams, mock-ups	Done
3 Database and prototype	Database design and script, clickable prototype	ER diagram, SQL script, prototype	Done, updated to v2.0
4 Authentication	Registration with CNIC and OTP, login, lockout, reset	M1 complete	Done
5 Student portal and services	Dashboard, apply form, service engine, uploads, edit and resubmit	M2 and M3 complete	Done
6 Workflow, staff, campuses	Requests, review, staff, roles, students, campus scope, Campus Admin	M4, M5 and M6 complete	Done
7 Reports, notifications, system	Dashboards, six reports, emails, PDF, settings, backup, PWA, chatbot	M7 to M11 complete	Done
8 Testing	Test cases, security tests, fixes, user trial with real staff	Test report	In progress
9 Deployment and documents	Hosting, backup routine, user manual, final report, viva preparation	Live system and final documents	In progress

10.3 Work split

Member	Focus
Muhammad Hussnain	Back-end: controllers, database, security, deployment.
Tehseen Qaiser	Front-end: pages, responsive layout, charts, PWA; interface testing.
Abdullah Ghaffar	Reports, notifications, documentation, test cases and user manual.

The split is indicative; every member also reviews the code of the others.

11 Testing Strategy

Testing happens at four levels: module tests while building, integration tests when modules are joined, security tests, and a user trial with a few students and staff.

ID	Scenario	Expected result
TC-01	Register with a valid CNIC, a campus and a department of that campus	OTP is emailed; account is created after the correct code
TC-02	Register with a department that belongs to another campus	Rejected with a clear message
TC-03	Register with a weak password such as 12345678	Rejected by the password policy
TC-04	Log in with a wrong password five times	Sign-in is locked and the wait time is shown
TC-05	Forgot password, correct code, new password	Password changed; old password no longer works
TC-06	Apply for a service with a missing required field	Form is refused and the field is named
TC-07	Apply twice for the same service while the first is open	Second application is refused
TC-08	Staff of campus A opens the requests list	Only campus A requests of allowed services appear
TC-09	Staff of campus A opens a campus B request by changing the number in the address	Access refused
TC-10	Staff limited to some statuses tries to reject a request	Action is not offered and is refused if forced
TC-11	Campus Admin opens Students and Reports	Only own campus data appears
TC-12	Return a request, then the student edits and resubmits	Status goes back to Pending with the new answers
TC-13	Complete a Wi-Fi request with credentials	Student receives the email; status shows Completed
TC-14	Chatbot: track by code, and by AG No. with a wrong CNIC	Code shows status; wrong CNIC returns "not found"; repeated tries are rate limited
TC-15	Open the site on a phone	Menu, tables, forms and chatbot are usable without sideways scrolling
TC-16	Create a backup, change data, restore the backup	Data returns to the backup state
TC-17	Switch maintenance mode on	Visitors see the maintenance page; Super Admin can log in

12 Deployment and Maintenance

- **Environment:** PHP 8.1+ and MySQL / MariaDB on a university server or shared hosting.
- **Install:** upload the files and open install.php. Choose Local (XAMPP) or Live, enter the database details; the installer creates the 13 tables, the system roles and the 7 services, creates the Super Admin with a generated strong password, writes config/local.php and deletes itself. A manual install uses database/schema.sql, database/seed.sql and scripts/create_superadmin.php.
- **After install:** enter SMTP details in Site Settings and send a test email, then add campuses and departments and any Campus Admins.

- **Backups:** recommended before every update and on a regular schedule (for example weekly); stored backups can be downloaded from Site Settings.
- **Updates:** database changes are delivered as upgrade scripts that can be run safely more than once.
- **Support:** the audit log and the server error log are the first place to look when something fails.

13 Risks and Mitigation

Risk	Impact	Mitigation
Offices change their approval process	Medium	Services, fields and roles are data, so they can change without code.
Email delivery problems	High	SMTP test, clear error messages, credentials shown to the administrator if sending fails.
Wrong campus or department data	Medium	Admin screens to correct them; upgrade scripts move old data safely.
Security weakness in uploads, sessions or login	High	Validation, strict cookies, lockout, security headers, regular security tests.
Chatbot used to guess student data	Medium	Both AG No. and CNIC must match; rate limits; no personal data in answers.
Time pressure before the final review	Medium	Core features are finished; remaining work is testing, manual and hardening.

14 Future Enhancements

- Online fee payment through a bank or wallet gateway, replacing the uploaded challan.
- SMS and push notifications.
- An approval chain with several desks in order.
- A student and staff mobile app with offline viewing of request status.
- Chatbot answers in Urdu.