

Project 17: Pioneerlinks - Education Consultant Management System

About the Client

Pioneerlinks Education & Visa Consultants in Melbourne are dedicated to assist and guide international students with selecting courses in various education sectors which are aligned with their career goals.

We also assist for the **family visa Australia** to those families who wanted to migrate to Australia. At Pioneerlinks Career Consultancy the education advice is complimented with professional migration advice to design a comprehensive career strategy. Pioneerlinks Migration team of dedicated professionals equipped with sound knowledge of Australian Education Industry, National Code and Australian immigration law providing informed advice to our clients maintaining highest satisfaction level.

At Pioneerlinks Migration, We have a highly experienced and registered migration consultants who provide their services to the clients throughout the world want to apply for Australian visa. We as a registered migration agent in Melbourne provide services and assist our clients through the complex process of visa application.

We will probably manage and give a consistent domain in the middle of customer and lawyer all through the movement process. We offer an open, available law office, willing to speak to your hobbies morally, ardently and speedily.

Our expert specialists are accessible to help with:

- Student visa applications,
- Business visa application, Visitor Visa
- Family stream applications including life partner/accomplice and guardian movement.
- Employer Nomination Visa, General Skilled Migration.
- Regional state sponsorships.
- MRT/AAT Appeals for refusal and Cancel Visa Services

Project Brief & Business Problem Specifications:

Client is facing the following problems and issues in the existing system:

Answering Questions

Hard to simplify Bookkeeping

Payment Problem Management

Organizing the Document Lifecycle

Information Management Operations

Hiring and Maintaining Quality Staff

Overall Responsiveness to Clients

Don't Know What Will Go Wrong

Lost documents

System Modules Requirements

1. User Management Module

- Student registration and login
- Consultant and admin account management
- Role-based access control (Student, Consultant, Admin)
- Password reset and account recovery
- Profile management with personal details

2. Student Profile Management Module

- Store academic history and qualifications
- Upload documents (transcripts, test scores, CV)
- Manage personal and contact details

- Track application history
- Maintain consultation notes and records

3. Consultant Profile Management Module

- Create and manage consultant accounts
- Define expertise areas and countries handled
- Upload profile photo and bio
- Manage availability schedule
- View assigned student list

4. Appointment Scheduling Module

- Students book consultations with consultants
- View available time slots in calendar
- Confirm, cancel, or reschedule appointments
- Send email/SMS confirmations and reminders
- Consultant view of daily/weekly schedule

5. Application Management Module

- Create student applications for target institutions
- Track application status (Draft, Submitted, Under Review, Accepted, Rejected)
- Upload and store application documents
- Record communication with institutions
- View application history

6. Course and Institution Database Module

- Maintain list of partner universities/colleges
- Store course details, eligibility, fees, deadlines
- Search and filter courses by criteria
- Update course and institution profiles
- Link courses to student applications

7. Document Management Module

- Secure upload of student documents
- Categorise and tag documents (e.g., Passport, Transcript)
- Download and share documents with institutions
- Maintain version history if updated
- Role-based document access control

8. Communication and Messaging Module

- Internal messaging between students and consultants
- Email/SMS notifications for appointment reminders
- Notify students about application updates
- Broadcast announcements or newsletters
- Store message history for compliance

9. Payment and Billing Module

- Create and manage invoices for consultation fees
- Accept online payments (credit card, bank transfer)
- Track payment status (Paid, Unpaid, Overdue)

- Issue digital receipts
- View payment history by student

10. Task and Workflow Management Module

- Assign tasks to consultants or admin staff
- Define task status (Pending, In Progress, Completed)
- Set task deadlines and reminders
- View personal and team task lists
- Log task completion history

11. Reporting and Analytics Module

- Generate reports on student registrations
- View consultant workload and appointments
- Track application success rates
- Financial reports for payments and invoices
- Export reports in PDF or CSV

12. Admin Dashboard and Management Module

- Manage user accounts and roles
- Oversee all appointments and applications
- Access system-wide analytics and reports
- Configure site settings and preferences
- View system logs and audit trails

13. Notifications Module

- Send email and SMS notifications for key events
- Appointment confirmations and reminders
- Application status updates
- Payment and invoice notifications
- Marketing communications with opt-out option

14. Content Management Module

- Manage website pages (About, Contact, Services)
- Post news or blog articles
- Upload promotional banners
- Edit FAQs and help content
- Configure SEO settings

15. Security and Audit Module

- User activity logging and audit trails
- Data encryption at rest and in transit
- Enforce strong password policies
- Role-based access permissions
- Alerts for suspicious activity

UI Design

User Interface for this system is concerned with the dialogue between a user and the computer. It is concerned with everything from starting the system or logging into the system to the eventually presentation of desired

inputs and outputs. The overall flow of screens and messages is called a dialogue.

UI Design Requirements

1. The system user should always be aware of what to do next.
2. The screen should be formatted so that various types of information, instructions and messages always appear in the same general display area.
3. Message, instructions or information should be displayed long enough to allow the system user to read them.
4. Use display attributes sparingly.
5. Default values for fields and answers to be entered by the user should be specified.
6. A user should not be allowed to proceed without correcting an error.
7. The system user should never get an operating system message or fatal error.

The aim of proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitations of the existing system. The system provides proper security and reduces the manual work.

- Security of data.
- Ensure data accuracy's
- Proper control of the higher officials.
- Minimize manual data entry.
- Minimum time needed for the various processing.
- Greater efficiency.
- Better service.
- User friendliness and interactive.

- Minimum time required.

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear for the stakeholders. Generally, functional requirements describe system behavior under specific conditions. The developers of this system must enhance the performance and efficiency of the system by adding all the required functional requirements. Students need to do their own research to find how they can improve the system and which FRs need to be added. The group must need a prior approval from the stakeholders/project supervisor before finalizing these Functional Requirements. These enhanced FRs must be reflected separately in Final SRS Report after the approval.

Functional Requirements

- FR1.** Allow students to register accounts with email verification.
- FR2.** Provide secure login and logout using PHP sessions.
- FR3.** Enable password reset and account recovery.
- FR4.** Let users edit personal profile details stored in MySQL.
- FR5.** Enforce role-based access control for Students, Consultants, and Admins.
- FR6.** Store student profiles including academic history, qualifications, and contact information.
- FR7.** Enable upload of documents (transcripts, ID, test scores) with secure PHP file handling.
- FR8.** Allow students to view and update their profile data.
- FR9.** Track student consultation history in the database.
- FR10.** Provide a student dashboard with profile summary.
- FR11.** Allow consultants to create and manage their profiles with expertise areas.
- FR12.** Store consultant details in MySQL including bio, photo, and availability.
- FR13.** Enable consultants to update their schedules.
- FR14.** Display consultant profiles to students for selection.
- FR15.** Assign students to consultants for ongoing case management.
- FR16.** Enable students to view consultant availability calendars.
- FR17.** Allow students to book consultation appointments online.
- FR18.** Store appointments in MySQL with date, time, and participant details.
- FR19.** Enable consultants to confirm, reschedule, or cancel appointments.
- FR20.** Send email/SMS confirmations and reminders using PHP integrations.
- FR21.** Allow consultants to create and manage student applications to institutions.
- FR22.** Store application details in the database with statuses (Draft, Submitted, Under Review, Accepted, Rejected).
- FR23.** Enable upload and storage of application documents.

- FR24.** Track communication notes between consultant and institution.
- FR25.** Provide students with application status tracking.
- FR26.** Maintain a searchable database of partner institutions and courses.
- FR27.** Store institution details including contact info, locations, and images.
- FR28.** Manage course listings with eligibility, fees, and deadlines.
- FR29.** Allow admins to add, edit, or delete institution and course records.
- FR30.** Link student applications to specific courses or institutions.
- FR31.** Allow secure upload of student documents (PDF, images) via PHP file handling.
- FR32.** Store document metadata in MySQL with categories and tags.
- FR33.** Enable role-based document access (Student, Consultant, Admin).
- FR34.** Support document versioning or history if updated.
- FR35.** Allow download and sharing of documents with institutions.
- FR36.** Provide internal messaging between students and consultants.
- FR37.** Store messages in MySQL with sender, receiver, and timestamp.
- FR38.** Notify users of new messages via email or SMS.
- FR39.** Enable announcements or newsletters to all users.
- FR40.** Log all communications for compliance and tracking.
- FR41.** Generate invoices for consultation fees.
- FR42.** Allow online payments using integrated payment gateways in PHP.
- FR43.** Store payment records with status (Paid, Unpaid, Overdue) in MySQL.
- FR44.** Issue digital receipts upon payment completion.
- FR45.** Enable students and admins to view payment history.
- FR46.** Allow consultants and admins to create tasks for student applications.
- FR47.** Store tasks in the database with status (Pending, In Progress, Completed).
- FR48.** Set deadlines and send reminders.
- FR49.** Allow assignment of tasks to specific staff members.
- FR50.** Display personal and team task lists on dashboard.
- FR51.** Generate reports on student registrations and profiles.
- FR52.** Provide consultant workload and appointment reports.
- FR53.** Track application success rates by consultant or institution.
- FR54.** Generate financial reports on payments and invoices.
- FR55.** Export reports in PDF or CSV format using PHP libraries.
- FR56.** Provide an admin dashboard with site-wide statistics.
- FR57.** Enable management of user accounts and roles.
- FR58.** Oversee appointments, applications, and payments.
- FR59.** Access audit logs and system settings.
- FR60.** Configure email/SMS notification templates.
- FR61.** Send email and SMS notifications for appointment confirmations and reminders.
- FR62.** Notify students about application status changes.
- FR63.** Send payment and invoice notifications.
- FR64.** Enable marketing announcements with unsubscribe options.
- FR65.** Log notification delivery and failures.

- FR66.** Manage static website pages (About, Services, Contact, FAQs).
- FR67.** Post news, blog articles, and announcements.
- FR68.** Upload and manage banners and images.
- FR69.** Allow scheduling of content publishing.
- FR70.** Configure SEO metadata for site pages.
- FR71.** Log all critical user actions for auditing (logins, edits, payments).
- FR72.** Store audit logs securely in MySQL.
- FR73.** Provide admins with audit log review tools.
- FR74.** Maintain logs for at least 12 months.
- FR75.** Support export of audit logs in CSV.

Non-Functional Requirements

| NFR ID | Category | Requirement Description |
|--------|--------------|---|
| NFR1 | Performance | Support at least 200 users with average page load times under 2 seconds. |
| NFR2 | Performance | Ensure responsive PHP server-side processing during peak usage. |
| NFR3 | Performance | Optimize MySQL queries with indexes to handle large datasets. |
| NFR4 | Performance | Enable efficient pagination and search for student profiles and applications. |
| NFR5 | Performance | Allow horizontal scaling of PHP servers behind a load balancer. |
| NFR6 | Security | Encrypt sensitive data at rest in MySQL (e.g., bcrypt-hashed passwords). |
| NFR7 | Security | Enforce HTTPS for all data transmission. |
| NFR8 | Security | Use prepared statements or ORM to prevent SQL injection. |
| NFR9 | Security | Implement secure PHP session management with timeouts and regeneration. |
| NFR10 | Security | Enforce role-based access control consistently in PHP code. |
| NFR11 | Security | Log all critical user activities (logins, payments, data edits) for auditing. |
| NFR12 | Availability | Maintain at least 99.5% system uptime excluding planned maintenance. |
| NFR13 | Availability | Provide clear maintenance notices for users on the site. |
| NFR14 | Availability | Support database replication or failover for high |

| NFR ID | Category | Requirement Description |
|---------------|-----------------|---|
| | | availability if required. |
| NFR15 | Reliability | Ensure transactional integrity in MySQL during critical operations. |
| NFR16 | Reliability | Enable daily automated database backups. |
| NFR17 | Reliability | Support point-in-time recovery for restoring system state after failure. |
| NFR18 | Reliability | Avoid data loss in confirmed transactions during server failures. |
| NFR19 | Usability | Deliver an user-friendly interface for all roles. |
| NFR20 | Usability | Ensure responsive design compatible with desktops, tablets, and smartphones. |
| NFR21 | Usability | Provide clear, human-readable validation errors and feedback. |
| NFR22 | Usability | Ensure accessibility compliance with WCAG 2.1 Level AA guidelines where feasible. |
| NFR23 | Compatibility | Support major browsers (Chrome, Firefox, Safari, Edge). |
| NFR24 | Compatibility | Ensure compatibility with PHP 7.4 or higher. |
| NFR25 | Compatibility | Support MySQL 5.7 or higher using InnoDB storage engine. |
| NFR26 | Compatibility | Ensure consistent behaviour across mobile and desktop devices. |
| NFR27 | Scalability | Use normalised MySQL schemas to reduce redundancy. |
| NFR28 | Scalability | Allow scaling of PHP servers using load balancers. |
| NFR29 | Scalability | Enable MySQL replication for read-heavy operations. |
| NFR30 | Scalability | Support modular addition of new features with minimal refactoring. |
| NFR31 | Maintainability | Follow PSR-12 PHP coding standards for consistency. |
| NFR32 | Maintainability | Use MVC or layered architecture for code separation and maintenance. |
| NFR33 | Maintainability | Document PHP code, MySQL schema, and APIs thoroughly. |
| NFR34 | Maintainability | Provide admin tools for managing users, content, applications, and payments. |
| NFR35 | Maintainability | Log and monitor PHP errors with clear diagnostic messages. |
| NFR36 | Audit and | Maintain audit logs of all sensitive user actions in |

| NFR ID | Category | Requirement Description |
|---------------|----------------------|--|
| | Compliance | MySQL. |
| NFR37 | Audit and Compliance | Retain audit logs for at least 12 months. |
| NFR38 | Audit and Compliance | Ensure data handling complies with privacy regulations (e.g., GDPR). |
| NFR39 | Audit and Compliance | Allow export of audit logs in CSV or other standard formats. |
| NFR40 | Backup and Recovery | Perform automated daily backups of the MySQL database. |
| NFR41 | Backup and Recovery | Allow on-demand manual backups initiated by administrators. |
| NFR42 | Backup and Recovery | Store backups securely with restricted access. |
| NFR43 | Backup and Recovery | Ensure recovery of full system within 4 hours of a critical failure. |
| NFR44 | Legal and Ethical | Display clear Terms of Service and Privacy Policy to all users. |
| NFR45 | Legal and Ethical | Obtain user consent for data collection and communication preferences. |
| NFR46 | Legal and Ethical | Provide users the ability to request account deletion and data removal in line with privacy regulations. |
| NFR47 | Legal and Ethical | Handle all personal, financial, and academic data ethically and securely. |

Hardware Requirement: Should be recommended by the developers.

Software Requirement: Should be recommended by the developers.