**There are 18 primary modules in this application:**

**1. Login**

* Secure user authentication (username & password).
* Validates user role (recruiter or applicant).
* Protected from SQL injection vulnerabilities using prepared statements in backend.

**2. Registration**

* Enables applicants to create accounts with personal details.
* Validates secure password policies.

**Applicant Side**

**3. Open Recruitments**

* Displays active recruitments with details and deadlines.
* Provides option to view full recruitment advertisement.

**4. Apply**

* Allows applicants to fill out application forms.
* Pre-filled inputs and necessary checks placed to ensure error-free user input.

**5. Closed Recruitments**

* Lists past recruitments for historical reference.
* Displays final appointment status (if available).

**6. Open Applications**

* Shows applicant’s active applications and current statuses.
* Allows cancellation of applications.

**7. Closed Applications**

* Displays applications where recruitment has closed.
* View final ranking and verification status.

**8. Open Offers**

* Displays active appointment letters issued by recruiters.
* Allows accepting/rejecting the offers.

**9. Previous Offers**

* Archives past appointment offers.
* Keeps record of accepted/rejected offers.
* Available for future reference.

**10. Profile**

* Allows applicants to update personal details and contact info.
* Uploads profile photo or other identification docs.

**Recruiter Side**

**11. Create Recruitment**

* Enables creation of new recruitment postings.
* Sets title, salary, deadlines, vacancies

**12. Add/Modify Subjects**

* Defines subjects relevant to recruitment result processing.
* Sets priority for each subject (weightage for final ranking).
* Editable until recruitment published.

**13. Manage Recruitments**

* Lists all recruitments created by the recruiter.
* Allows editing of active recruitments.

**14. Submit Results**

* Enables recruiters to enter subject-wise marks for applicants.
* Validates marks entry against recruitment subject priorities.
* Secure storage of entered marks in the database.

**15. Create Rank List**

* Generates final applicant ranking based on aggregate scores.
* Considers subject priorities and date of birth as tiebreakers.
* Automatically updates recruitment records with rank data.

**16. Verify Applications**

* Allows recruiters to verify application details and documents.
* Marks applications as verified/rejected
* Prepares data for appointment offer issuance.

**17. Send Appointments**

* Generates appointment letters for verified applicants.
* Attaches secure document hash for tracking and download.
* Allows sending appointments to verified applicants in the scheduled category against which vacancies exist only.

**18. Manage Appointments**

* Lists all issued appointment letters and their statuses.
* Tracks applicant responses.

**Data Structures**

All modules use a mix of Javascript objects, arrays, react state variables on the front end and JSON for communication between the React/js front-end and PHP backend. The Backend uses a mix of associative and non-associative arrays.

**Data Integrity**

**1. Referential Integrity**

* Foreign keys maintain relationships between tables, ensuring consistent data across the system.
* ON DELETE ON UPDATE constraints for automated update of dependent records (e.g., removing recruitments automatically removes related applications and results).

**2. Validation Constraints**

* Field-level constraints (e.g., ENUMs, numeric ranges) ensure valid data storage.
* Unique constraints on critical fields (e.g., username, document hash) prevent duplicates.
* Date and time fields are validated to match expected formats.

**4. Consistent Ranking Logic**

* Aggregate scores and priority-based calculations are standardized for fair ranking.
* Tiebreakers (date of birth) ensure no duplicate ranks for final appointments.

**5. Secure File Handling**

* Uploaded documents use hashed filenames and are tied to respective application/recruitment records.
* File metadata (type, hash, original name) is stored safely and linked to document records.

**6. Controlled Access**

* Role-based access control prevents unauthorized operations (e.g., applicants can’t modify results).
* Secure authentication and authorization layers for data access.

**7. User Input Checks**

* **Frontend validation** for input types (numbers, text, file types) minimizes incorrect entries.
* **Server-side checks** on all user inputs to validate data length, format (e.g., emails, names), and file size/type.
* Inline form feedback (highlighting errors and guiding users to correct them) reduces invalid submissions.