**List of Modules and Features**

1. **Job Seekers Module** (Focus: Self-service tools for job search and career development)
   * Seamless lifecycle management (intake to case closure).
   * User registration, profile creation/editing (including resumes and cover letters).
   * Job application tracking with real-time status dashboards.
   * Intelligent job matching based on skills, history, and education.
   * Access to labor market data, skills gap analysis, and forecasts.
   * Self-scheduling of appointments and assessments.
   * Alerts for job matches, deadlines, and missing profile info.
   * Veteran/spouse notifications and resource referrals.
   * AI-assisted resume building and job searches.
   * Upload of program activity documentation (e.g., SNAP E&T, WT/TANF).
2. **Employers Module** (Focus: Recruitment and talent acquisition)
   * Employer registration, profile management, and approval workflows.
   * Job posting/editing (including on-the-job training).
   * Full hiring lifecycle management (posting to fulfillment).
   * Candidate search (manual/automated) with filters (e.g., geography, skills).
   * Visibility into tax incentives, program eligibility flags on resumes.
   * Labor market data access and talent availability assessments.
   * Participation in special initiatives (e.g., Hope Florida).
   * AI for job posting generation and candidate screening.
   * Option for anonymous job postings.
3. **Workforce Staff Module** (Focus: Case management and administrative tools)
   * End-to-end case management (intake, eligibility, services, follow-up).
   * Registration of job seekers, employers, and providers.
   * Filtering/ranking of job seekers and job orders.
   * Resume flagging for program eligibility.
   * Customizable dashboards for case status, actions, and notes.
   * Service tracking, compliance monitoring, and voucher issuance.
   * Event creation and participant registration.
   * Eligibility determination and plan development (with templates).
   * AI for case note writing and searchability.
4. **Providers Module** (Focus: Training and education provider management)
   * Self/staff-assisted registration and profile maintenance.
   * Portal for Eligible Training Provider List (ETPL) applications under WIOA.
   * Document submission, approval/denial workflows, and public repository.
   * Periodic reviews for compliance reaffirmation.
5. **Labor Market Information Module** (Focus: Data-driven insights)
   * Display/filtering of current labor market data from Commerce's Bureau of Workforce Statistics and Economic Research (WSER).
   * Employer identification (local/multi-location) via geographic search.
   * Labor market forecasts and economic data access.
   * Skills gap analysis and job-specific requirements.
6. **Labor Exchange Module** (Focus: Job matching and aggregation)
   * Integration of employer-submitted and external job postings.
   * Duplicate detection and prevention.
   * Maximization of active/open job listings.
7. **Case Management Module** (Focus: Participant tracking and compliance)
   * Collection of personal/demographic data for eligibility (e.g., name, SSN, income, veteran status).
   * Automated service tracking using standardized codes.
   * Staff time tracking and funding stream association.
   * Interactive calendars for scheduling.
8. **Business Operations Module** (Focus: Core administrative and workflow tools)
   * Satisfaction surveys and reporting.
   * Automation of manual processes (e.g., e-signatures, auto-populating forms).
   * Financial tracking (obligations, expenditures, vouchers).
   * Multilingual support and custom geographic areas.
   * Petition tracking (Florida/nationwide).
   * Mass mailing generation.
   * CRM integration (from CareerSource Florida Salesforce).
   * Support for LWDB services and rapid deployment for urgent events.
9. **Reporting Module** (Focus: Compliance and analytics)
   * Customizable ad-hoc reports, dashboards, gauges, and graphs.
   * Federal/state/local reporting (e.g., quarterly/annual submissions).
   * Real-time/historical/auto-scheduled reports.
   * Participant data tracking for compliance.
   * Aggregated job/applicant trend analysis.
10. **Organizational Change Management (OCM) Module** (Focus: Training and adoption)
    * OCM Plan development for training needs.
    * Creation of training materials and Train-the-Trainer programs.
    * Instructor-led training delivery.
    * Coordination with Department's OCM team.
11. **Environment and Operations & Maintenance (O&M) Module** (Focus: Infrastructure and support)
    * 24/7 availability (excluding scheduled downtime).
    * Disaster recovery and business continuity planning.
    * Change management, issue tracking, and notifications.
    * Development/pre-production/training environments mirroring production.
12. **Data Management Module** (Focus: Storage, validation, and governance)
    * Data dictionary maintenance.
    * On-demand database access.
    * Data validation, encryption, and archiving.
    * Duplicate record prevention.
    * Compliance with retention policies.
13. **Document Management Module** (Focus: Content handling)
    * Support for various formats (PDF, Word, images).
    * Storing, indexing, versioning, access control, and reporting.
    * Scalable capacity for digital content (e.g., emails, faxes, OCR data).
14. **Integration Module** (Focus: Connectivity with external systems)
    * Interfaces with systems like Reconnect, FLORIDA System, FL WINS, Department of Education, CLIFF Tool.
    * API/file transfers, event-driven orchestration.
    * Error handling and logging.
15. **Security Module** (Focus: Protection and compliance)
    * Role-based access, multi-factor authentication.
    * Vulnerability management, encryption (FIPS 140-2+).
    * FedRAMP ATO (Moderate+), SOC 2 Type II audits.
    * Audit logs, geolocation controls, data masking.
16. **Testing Module** (Focus: Quality assurance)
    * Test planning (unit, integration, regression, performance, UAT).
    * Defect tracking and traceability to requirements.
    * Support for Department-led UAT.
17. **Transition Module** (Focus: Deployment and handover)
    * Minimized impact migration from legacy systems.
    * Documentation, training, and monitoring during initial operations.

**Data Analysis and Integration Areas**

Yes, data analysis and integration are major emphases in the RFP.

* **Data Analysis:**
  + Robust reporting and analytics (real-time dashboards, ad-hoc reports, business intelligence tools).
  + AI integration for skills gap analysis, job matching, trend analysis, and predictive insights (e.g., labor market forecasts, participant outcomes).
  + Centralized data warehouse for aggregating structured/semi-structured data from multiple sources.
  + Compliance tracking (e.g., performance metrics, federal/state reporting).
  + Usage metrics and satisfaction surveys for strategic decision-making.
* **Integration Areas:**
  + Modernized APIs and file transfers with systems like Reconnect, FLORIDA System, FL WINS, Department of Education, Department of Revenue, State College of Florida, CareerSource Florida Salesforce, and WSER databases.
  + Data migration from legacy systems (e.g., OSST, reporting databases) with cleansing, validation, and metadata preservation.
  + Event-driven orchestration, error handling, and versioning for seamless data exchanges.
  + Integration with IAM/SSO tools, web services, queues, and message protocols.
  + Data governance (e.g., master data management, lifecycle management, quality controls) to resolve conflicts and ensure accuracy.

Based on the extensive modules and features from the provided WITS document, here is a plan for the following deliverables:

**1. Detailed Diagrams**

**a) High-Level System Architecture Diagram**

* Layered view showing:
  + User Interface (Job Seeker, Employer, Workforce Staff, Providers)
  + Application Services (Modules: Job Seekers, Employers, Case Management, Labor Market Info, Reporting, etc.)
  + Integration Layer (API gateway, Event bus, Connectors to external systems)
  + Data Management Layer (Transactional DB, Data Warehouse, Document Store)
  + Security Layer (IAM, Encryption, Access Control)
  + Operations & Maintenance (Monitoring, Disaster recovery)

**b) Module Interaction (Component) Diagram**

* Illustrate core module interactions, e.g.:
  + Job Seekers Module connects with Labor Market Info Module and Labor Exchange.
  + Workforce Staff Module using Case Management Module and Reporting Module.
  + Integration Module handling data exchange between external systems and internal modules.

**c) Data Flow Diagram (DFD)**

* For key processes such as:
  + Job seeker registration, profile update, and job application submission.
  + Employer posting a job and candidate screening.
  + Caseworker managing participant eligibility and service tracking.
  + Labor Market Information aggregation and presentation.

**d) Entity Relationship Diagram (ERD) / Data Model**

* Core entities:
  + User (with roles: Job Seeker, Employer, Staff, Provider)
  + Job Posting & Application
  + Case/Participant
  + Training Provider
  + Labor Market Data Record
  + Service & Compliance Record
  + Transaction Records (e.g., vouchers, payments)
  + Document Metadata

**2. Data Models**

* Define key tables and relationships such as:

| **Entity** | **Key Attributes** | **Relationships** |
| --- | --- | --- |
| User | UserID, Name, Role, Contact Info, Credentials | One-to-many with Job Applications, Cases |
| Job Posting | JobID, EmployerID, Title, Description, Status | Many-to-one Employer |
| Job Application | ApplicationID, UserID, JobID, Status, Date | Many-to-one Job Posting, User |
| Case | CaseID, UserID, Status, Eligibility, Services | One-to-many Service Records |
| Training Provider | ProviderID, Name, Address, ApprovalStatus | Linked to Training Programs |
| Labor Market Data | DataID, JobCategory, Region, Demand Forecast | Used for matching and reporting |
| Document | DocumentID, LinkedEntityID, Type, Date, Status | Polymorphic link to any entity |

* Include indexing strategies for performance and GDPR/Federal compliance on PII data.

**3. Implementation Timelines**

**Phase 1 - Requirements and Design (2 months)**

* Finalize detailed requirements.
* Create detailed diagrams and data models.
* Define security and compliance plans.

**Phase 2 - Core Module Development (4 months)**

* Job Seekers Module, Employers Module.
* Workforce Staff Module and Case Management.
* Integration and Security Modules foundation.

**Phase 3 - Provider & Labor Market Information Modules (3 months)**

* Providers Module, Training Provider workflows.
* Labor Market Information Module development.

**Phase 4 - Labor Exchange and Reporting Modules (2 months)**

* Job aggregation and duplicate prevention.
* Reporting dashboard and compliance reporting.

**Phase 5 - Operations & Maintenance and Support Modules (2 months)**

* Environment & O&M, Testing, Transition modules.
* Document Management and Data Management.

**Phase 6 - System Integration, Testing, and Pilot Rollout (3 months)**

* API connections to external systems.
* Security audits, performance tuning.
* Pilot with limited geography/user base.

**Phase 7 - Full Production Deployment and Ongoing Maintenance**

* Gradual scale-up with continuous enhancement.
* User training and organizational change management.

**Visualization Prep**

I can help create the following visual assets:

* High-level system architecture diagram (layered)
* Module/component interaction diagram
* Data Flow Diagram (DFD) for user onboarding & job application
* Entity Relationship Diagram (ERD) for main data entities
* Gantt chart for implementation timeline phases

Please confirm which diagrams you want first or if you want me to start with the architecture diagram and timeline chart.

1. <https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/attachments/84729513/762e92b0-b5c1-4693-9afc-cf73be758532/WITS-Modules-and-Features.docx>

**Screen Designs**

Based on the "Job Seekers Module" screens designed earlier, here’s a detailed list of **fields on each screen**, **validations**, and **business logic**:

## 1. User Registration & Profile Creation Screen

## Fields:

* First Name\* (text)
* Last Name\* (text)
* Date of Birth\* (date)
* Email\* (email)
* Phone Number\* (phone)
* Address (multi-line text)
* Username\* (text)
* Password\* (password)
* Confirm Password\* (password)
* Security Question\* (dropdown)
* Security Answer\* (text)
* Education (multi-choice dropdown/list)
* Work History (free text and structured entries)
* Skills (multi-select dropdown)
* Certifications (file upload or manual entry)
* Upload Resume\* (file upload: PDF/DOCX, max 10MB)
* Upload Cover Letter (optional file upload)
* Veteran Status (checkbox)
* Spouse Status (checkbox)
* Resource Referral Opt-in (checkbox)

## Validations:

* Required fields (\*) must be completed.
* Email to have valid email format, uniqueness checked in system.
* Password minimum 8 characters, mix of letters, numbers, special chars.
* Confirm Password must match Password.
* Phone number matches valid pattern; optional country code.
* Date of Birth validates minimum age (e.g., 16 years).
* File uploads must be of valid type and within size limits.
* Username uniqueness check.
* Security Answer cannot be blank.

## Business Logic:

* Upon registration, send verification email with confirmation link.
* Password stored using secure hashing.
* Resume uploads linked to user profile, with version control for updates.
* Veteran/spouse check triggers tailored resource notifications.
* Profile completion status tracked; reminders sent for missing sections.
* AI-assisted resume builder suggests improvements post-upload.

## 2. Job Application Tracking Dashboard

## Fields (display only, with filters on top):

* Job Title
* Employer Name
* Application Date
* Application Status (Received, Under Review, Interview Scheduled, Offered, Rejected)
* Location
* Job Type (Full-time, Part-time, Contract)
* Filters: Date range, Status, Location, Type

## Validations:

* Filter inputs validated for date ranges and enumerations.
* Dashboard auto-refresh every X minutes or manual refresh.

## Business Logic:

* Status updates pushed from employer side feed into system.
* Notifications generated on status changes (e.g., interview scheduled).
* Pagination for large application lists.
* Historical applications archived but accessible.
* Alerts highlight expiring job applications or required follow-ups.

## 3. Intelligent Job Matching Screen

## Fields:

* Search Bar (keywords)
* Filters: Location, Salary Range, Industry, Job Type
* Recommended Jobs list:
  + Job Title
  + Employer
  + Match Score (0-100%)
  + Short Description
  + Posted Date
  + Save Job (action button)
  + Apply (action button)

## Validations:

* Keyword length limit for search (e.g., 50 chars).
* Salary range must be numeric and valid (min ≤ max).
* Filter selections validated for permitted values.

## Business Logic:

* Match Score calculated dynamically based on user skills, history, and job requirements.
* AI engine updates recommendations daily.
* Users receive alerts on new job matches.
* Saving a job saves it to user’s profile and triggers reminders.
* Application process pre-fills data from user profile.

## 4. Self-Scheduling Appointments and Assessments Screen

## Fields:

* Appointment Type\* (dropdown: Career Counseling, Skills Assessment, Job Fair, etc.)
* Date Picker\* (disabled past dates)
* Time Slot Selector\* (shows available slots)
* Additional Notes (optional text)

## Validations:

* Required fields validation.
* Appointment date must be in future.
* Validation against overlapping appointments or max bookings per day.

## Business Logic:

* Available slots reflect current capacity and are real-time updated.
* Confirmation email/SMS sent upon booking.
* Reminders scheduled 24 hours and 1 hour before appointment.
* No-show and reschedule logic handled by staff module.

## 5. Alerts and Notifications Center

## Fields:

* Notification List:
  + Date
  + Type (Job Match, Deadline, Profile Alert, Veteran Resource)
  + Message preview
  + Status (Read/Unread)

## Validations:

* Pagination controls for large notification sets.
* Filter by date range and type.
* Mark as Read toggle button.

## Business Logic:

* New alerts generated by system events (job matches, profile incompleteness, deadlines).
* Settings allow user to choose notification channels.
* Actions embedded in alerts link to relevant screens (e.g., profile update).

## 6. Program Activity Documentation Upload Screen

## Fields:

* Program Type\* (dropdown: SNAP E&T, WT/TANF, Other)
* Upload Button (accepts PDF, DOCX, JPG, PNG)
* Document Description (text)
* Upload Date (auto-filled)
* Status (Submitted, Approved, Rejected)
* Notes (readonly, from reviewers)

## Validations:

* File type and size limitations.
* Required program type and description.
* Duplicate check for same document type and date range.

## Business Logic:

* Uploaded documents linked to user and program.
* Auto-notification to reviewers upon upload.
* Status updated by staff after review.
* Version history retained for corrections and re-submissions.

If desired, these can be extended into detailed UI wireframes and API specifications next.

1. <https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/attachments/84729513/762e92b0-b5c1-4693-9afc-cf73be758532/WITS-Modules-and-Features.docx>

Certainly! Below are the detailed UI wireframe descriptions and API specifications for the **Job Seekers Module** screens previously outlined. These should guide the front-end and back-end development, enabling smooth implementation.

# Job Seekers Module – UI Wireframes & API Specifications

## 1. User Registration & Profile Creation Screen

## UI Wireframe Description:

* Form layout with grouped sections: Personal Info, Login Credentials, Profile Details, Uploads.
* Validation messages displayed inline near fields.
* Progress indicator showing profile completeness.
* Save and Submit buttons at bottom.
* Responsive design for mobile/tablet.

## Key API Endpoints:

| **Endpoint** | **Method** | **Payload** | **Response** | **Description** |
| --- | --- | --- | --- | --- |
| /api/users/register | POST | { firstName, lastName, dob, email, phone, address, username, password, securityQuestion, securityAnswer } | { success, userId, message } | Register new user & generate verification token |
| /api/users/profile | GET | { userId } | { profileData } | Retrieve user profile data |
| /api/users/profile | PUT | { userId, education, workHistory, skills, certifications, veteranStatus, spouseStatus } | { success, updatedFields } | Update user profile |
| /api/users/uploadResume | POST | multipart/form-data { userId, file } | { success, fileUrl } | Upload resume file |
| /api/users/uploadCover | POST | multipart/form-data { userId, file } | { success, fileUrl } | Upload cover letter |

## 2. Job Application Tracking Dashboard

## UI Wireframe Description:

* Table listing jobs with sortable columns: Title, Employer, Date, Status.
* Filters panel on side or top.
* Notification badge for updates.
* Clickable rows open detailed application info.

## Key API Endpoints:

| **Endpoint** | **Method** | **Payload** | **Response** | **Description** |
| --- | --- | --- | --- | --- |
| /api/applications | GET | { userId, filters, page, size } | { applications[], total } | Fetch user’s job applications |
| /api/applications/{id} | GET | - | { applicationDetails } | Get detailed application info |
| /api/notifications | GET | { userId } | { notifications[] } | Fetch user notifications |

## 3. Intelligent Job Matching Screen

## UI Wireframe Description:

* Search bar with filter dropdowns.
* List of matched jobs with match score and apply/save buttons.
* Pagination and sorting options.
* Highlight new or highly matched jobs.

## Key API Endpoints:

| **Endpoint** | **Method** | **Payload** | **Response** | **Description** |
| --- | --- | --- | --- | --- |
| /api/jobs/matches | GET | { userId, keywords, filters, page } | { matchedJobs[], total } | Fetch intelligent job matches |
| /api/jobs/{id} | GET | - | { jobDetails } | Get job posting details |
| /api/jobs/save | POST | { userId, jobId } | { success } | Save job to user favorites |
| /api/jobs/apply | POST | { userId, jobId } | { success, applicationId } | Start job application process |

## 4. Self-Scheduling Appointments and Assessments Screen

## UI Wireframe Description:

* Calendar view with selectable dates/time.
* Appointment type dropdown.
* Confirmation summary page.
* Email and SMS reminders toggles.

## Key API Endpoints:

| **Endpoint** | **Method** | **Payload** | **Response** | **Description** |
| --- | --- | --- | --- | --- |
| /api/appointments/slots | GET | { appointmentType, dateRange } | { availableSlots[] } | Get available time slots |
| /api/appointments/schedule | POST | { userId, appointmentType, slotId, notes } | { success, appointmentId } | Schedule an appointment |
| /api/appointments/user | GET | { userId } | { appointments[] } | Fetch user’s appointments |

## 5. Alerts and Notifications Center

## UI Wireframe Description:

* List of notifications with marker for read/unread.
* Filter and search options by type, date.
* Action links embedded.
* Settings panel for notification preferences.

## Key API Endpoints:

| **Endpoint** | **Method** | **Payload** | **Response** | **Description** |
| --- | --- | --- | --- | --- |
| /api/notifications | GET | { userId, filters } | { notifications[] } | Fetch notifications list |
| /api/notifications/read | POST | { notificationId } | { success } | Mark notification as read |
| /api/notifications/settings | GET | { userId } | { settings } | Get user notification preferences |
| /api/notifications/settings | POST | { userId, settings } | { success } | Update notification preferences |

## 6. Program Activity Documentation Upload Screen

## UI Wireframe Description:

* Upload form with program type dropdown.
* File drag-and-drop or selective upload widget.
* List of uploaded documents with status and action buttons.
* History log for document submissions.

## Key API Endpoints:

| **Endpoint** | **Method** | **Payload** | **Response** | **Description** |
| --- | --- | --- | --- | --- |
| /api/documents/upload | POST | multipart/form-data { userId, programType, file, description } | { success, documentId } | Upload program documentation |
| /api/documents/list | GET | { userId, programType } | { documents[] } | List uploaded documents |
| /api/documents/status | GET | { documentId } | { status, reviewerNotes } | Check document review status |
| /api/documents/delete | DELETE | { documentId } | { success } | Delete/retract document |

Certainly! Below is a comprehensive deliverable package including:

1. **Detailed Database Schema Design** for the Job Seekers Module
2. **API Security Considerations** best practices
3. **UI/UX Visual Wireframe Descriptions** for key Job Seekers Module screens

# 1. Detailed Database Schema Design (Job Seekers Module)

## Entity Tables & Sample Fields

| **Table Name** | **Description** | **Key Fields** |
| --- | --- | --- |
| **Users** | Stores basic user info | UserID (PK), FirstName, LastName, DOB, Email, Phone, Address, VeteranStatus, SpouseStatus, CreatedAt, UpdatedAt |
| **UserCredentials** | Authentication details | CredentialID (PK), UserID (FK), Username, PasswordHash, Salt, SecurityQuestion, SecurityAnswerHash, CreatedAt |
| **Profiles** | Detailed user profiles | ProfileID (PK), UserID (FK), Education, WorkHistory, Skills (JSON), Certifications (JSON), Bio, UpdatedAt |
| **Resumes** | Resume storage and metadata | ResumeID (PK), UserID (FK), FilePath, FileType, UploadDate, IsActive |
| **CoverLetters** | Cover letter files | CoverLetterID (PK), UserID (FK), FilePath, FileType, UploadDate, IsActive |
| **Jobs** | Job postings | JobID (PK), EmployerID (FK), Title, Description, Location, SalaryRange, PostedDate, ExpiryDate, Status |
| **JobApplications** | Job application tracking | ApplicationID (PK), UserID (FK), JobID (FK), ApplicationDate, Status, StatusUpdateDate |
| **JobMatches** | Intelligent matching records | MatchID (PK), UserID (FK), JobID (FK), MatchScore, MatchDate, IsSaved |
| **Appointments** | Scheduled appointments | AppointmentID (PK), UserID (FK), Type, Date, TimeSlot, Status, CreatedAt, UpdatedAt |
| **Notifications** | User alerts and notifications | NotificationID (PK), UserID (FK), Type, Message, IsRead, CreatedAt |
| **ProgramDocuments** | Program-related doc tracking | DocumentID (PK), UserID (FK), ProgramType, FilePath, FileType, UploadDate, Status, Notes |

## Additional Notes:

* Use **foreign key constraints** to maintain relational integrity.
* Leverage **JSON fields** for flexible skill and certification arrays.
* Implement **soft deletes** with flag fields (IsActive).
* Index on frequently queried fields: UserID, JobID, ApplicationDate.

# 2. API Security Considerations

## Authentication & Authorization:

* Use **OAuth 2.0 / OpenID Connect** for secure, standardized authentication flows.
* Implement **JWT tokens** with short expiry for stateless API sessions.
* Enforce **role-based access control (RBAC)** to restrict endpoints by user roles.
* Mandatory **multi-factor authentication (MFA)** especially for sensitive operations.

## Secure Data Transmission:

* Enforce **TLS 1.2+** for all API endpoints.
* Use **HTTP Strict Transport Security (HSTS)** headers.

## Input Validation & Rate Limiting:

* Validate all user inputs server-side to avoid injection attacks.
* Implement **API throttling and rate limiting** to prevent DDoS and abuse.

## Data Privacy & Compliance:

* Encrypt PII at rest using **FIPS 140-2 compliant** encryption standards.
* Follow **data minimization principles**, only expose necessary data in payloads.
* All audit-relevant actions should be logged and monitored.

## Error Handling:

* Return **generic error messages** to clients; log detailed errors internally.
* Use standardized **HTTP status codes** (401 Unauthorized, 403 Forbidden, 400 Bad Request).

## API Gateway & Monitoring:

* Employ an API Gateway for request routing, authentication, and security policies.
* Real-time monitoring with alerts on anomalies or security incidents.

# 3. UI/UX Visual Wireframe Descriptions

## 3.1 User Registration & Profile Creation

* Form-based layout, divided into sections: Contact Info, Credentials, Profile Details, Document Upload.
* Inline validation (e.g., invalid email, missing fields).
* Progress bar indicating profile completion.
* Upload widgets for resume & cover letters with drag-drop support.
* Mobile-responsive and accessible design.

## 3.2 Job Application Tracking Dashboard

* Table/grid with sortable columns: Job Title, Employer, Date Applied, Status.
* Top filter bar for status, location, and date ranges.
* Notification badges on updated applications.
* Bulk action buttons: withdraw application, view details.

## 3.3 Intelligent Job Matching Screen

* Search box with filter dropdowns on left.
* Main pane with job cards showing title, employer, salary, match %.
* Buttons to “Save” or “Apply” on each card.
* Sidebar with labor market insights and skill gap visualizations.

## 3.4 Self-Scheduling Appointments and Assessments

* Interactive calendar with selectable days.
* Time slots list for each day with availability.
* Appointment type dropdown.
* Confirmation modal summarizing selection before submission.

## 3.5 Alerts and Notifications Center

* List-style view grouped by notification type.
* Read/unread icons.
* Action buttons embedded (Update Profile, View Job, Contact Support).
* Settings option to toggle notification preferences.

## 3.6 Program Activity Documentation Upload

* Single-page form with dropdown for program type.
* File upload section supporting multiformats, with size limits and progress bar.
* Table displaying uploaded documents and status.
* Edit and delete options with confirmation dialogs.

If you want, I can also provide mockup visuals to accompany these descriptions or API specifications with example requests and responses.

1. <https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/attachments/84729513/762e92b0-b5c1-4693-9afc-cf73be758532/WITS-Modules-and-Features.docx>

Certainly! Below are conceptual **mockup visual descriptions** for the key Job Seekers Module screens, plus detailed **API specifications** with example requests and responses for each function.

**1. Mockup Visual Descriptions (Conceptual)**

**1.1 User Registration & Profile Creation Screen**

* **Layout:** Multi-step form with a progress bar at top: Step 1 - Personal Info, Step 2 - Credentials & Security, Step 3 - Profile Details, Step 4 - Upload Documents.
* **Key UI Elements:** Text fields, drop-downs, file upload widget, checkboxes (Veteran/Spouse), inline validation messages.
* **User Flow:** Users fill each section, receive field-specific feedback, and navigate with Next/Back buttons before final Submit.

**1.2 Job Application Tracking Dashboard**

* **Layout:** Data table with sortable headers: Job, Employer, Date Applied, Status, Actions.
* **Key UI Elements:** Filters at top (date range, status), pagination, notification badges for updates, action buttons (View, Withdraw).
* **User Flow:** Click row to open detailed status and timeline for application progress.

**1.3 Intelligent Job Matching Screen**

* **Layout:** Left panel filters (keywords, location, salary), main grid showing job cards.
* **Key UI Elements:** Job card includes title, employer, match %, brief description, “Save” and “Apply” buttons.
* **User Flow:** Users browse matches, save favorites, or proceed to application in a modal.

**1.4 Self-Scheduling Appointments and Assessments**

* **Layout:** Calendar view with available dates highlighted, selectable time slots below.
* **Key UI Elements:** Appointment type dropdown, confirmation modal popup, notification setup switches.
* **User Flow:** Select appointment type → pick date/time → confirm → receive confirmation message and email.

**1.5 Alerts and Notifications Center**

* **Layout:** Scrollable list grouped by date with icon representing alert type.
* **Key UI Elements:** Mark as read/unread, filter by type, clickable links within alerts.
* **User Flow:** Users review alerts, click to navigate to relevant action screens.

**1.6 Program Activity Documentation Upload Screen**

* **Layout:** Form with dropdown for program type, drag/drop file upload area.
* **Key UI Elements:** Upload progress bar, document list table with statuses, edit/delete action buttons.
* **User Flow:** Upload document → View list with status → Option to replace or add notes.

**2. API Specifications with Example Requests & Responses**

**2.1 User Registration API**

**POST** /api/users/register

**Request Body:**

json

{

"firstName": "John",

"lastName": "Doe",

"dob": "1990-05-25",

"email": "john.doe@example.com",

"phone": "123-456-7890",

"address": "123 Main St, Cityville",

"username": "johndoe90",

"password": "StrongPass!123",

"securityQuestion": "Mother's maiden name?",

"securityAnswer": "Smith"

}

**Response:**

json

{

"success": true,

"userId": "abc123xyz",

"message": "Registration successful. Verification email sent."

}

**2.2 Profile Update API**

**PUT** /api/users/profile

**Request Body:**

json

{

"userId": "abc123xyz",

"education": "B.S. Computer Science",

"workHistory": "3 years at TechCo as Developer",

"skills": ["Java", "SQL", "AWS"],

"certifications": ["AWS Certified Developer"],

"veteranStatus": true,

"spouseStatus": false

}

**Response:**

json

{

"success": true,

"updatedFields": ["education", "workHistory", "skills", "certifications"]

}

**2.3 Job Applications List API**

**GET** /api/applications?userId=abc123xyz&page=1&size=10&status=Under Review

**Response:**

json

{

"applications": [

{

"applicationId": "app789",

"jobId": "job456",

"jobTitle": "Software Engineer",

"employerName": "TechCo",

"applicationDate": "2025-05-01",

"status": "Under Review"

}

],

"total": 1

}

**2.4 Intelligent Job Match API**

**GET** /api/jobs/matches?userId=abc123xyz&keywords=developer&location=Cityville&page=1

**Response:**

json

{

"matchedJobs": [

{

"jobId": "job456",

"title": "Software Engineer",

"employer": "TechCo",

"location": "Cityville",

"matchScore": 92,

"postedDate": "2025-04-28"

}

],

"total": 1

}

**2.5 Appointment Scheduling API**

**POST** /api/appointments/schedule

**Request Body:**

json

{

"userId": "abc123xyz",

"appointmentType": "Career Counseling",

"slotId": "slot234",

"notes": "Need help with resume."

}

**Response:**

json

{

"success": true,

"appointmentId": "appt567",

"message": "Appointment scheduled successfully."

}

**2.6 Upload Program Document API**

**POST** /api/documents/upload

**Request:** Multipart form-data with fields:

* userId=abc123xyz
* programType=SNAP E&T
* description=Verification document
* file=[binary file]

**Response:**

json

{

"success": true,

"documentId": "doc890"

}

These descriptions and API specs provide a strong foundation for development teams to implement the Job Seekers Module UI and backend with secure and efficient data handling. Wireframes can be created next if you want visual mockups in diagram form.

1. <https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/attachments/84729513/762e92b0-b5c1-4693-9afc-cf73be758532/WITS-Modules-and-Features.docx>