**Scope document for Vigyana web based Application**  
**Version: 1.0**  
**Date: 10, March 2025**

**Created By: Espoir Technologies Private Limited**

**Part 1**

**1. Purpose**

Develop a web application (Vigyana – AI Tool for Research) for higher education institutions and individuals to generate customized reports/solutions, based on their various inputs, using a credit-based system – A standalone fully working application is the base for web development.

**2. Scope**

* **Target Users**:
  + **Individual Users**: Students, researchers, or professionals who register, purchase credits, and generate reports.
  + **Institutional Users**: Universities, colleges, or organizations managed by a Super Admin who distributes credits to users.
  + **Guest User**
  + **Espoir Admin**
* **Core Features**:
  + User registration and authentication.
  + Credit purchase and management.
  + ChatGPT integration.
  + Research Report generation and credit consumption (Existing Vigyana Application explained in Part 2 Document with screenshots)
  + User dashboards for report history and downloads.
  + Super Admin dashboard for institutional credit/user management.
  + Espoir dashboard for overall administration.

**3. Functional Requirements**

**3.1 User Registration & Authentication**

* **Guest Users:**
  + Registration with valid email (no payment required).
  + Email verification via OTP or confirmation link.
  + 3 free credits granted automatically after email verification.
  + Cannot purchase additional credits unless upgraded to Individual User.
  + Profile management limited to email and password updates.
  + Guest can attempt only once. Ie One email-ID can gen 3 reports only.
* **Individual Users:**
  + Self-registration via email password or OAuth(Google/Microsoft)
  + Profile management (name, contact, profession, password reset)
  + Option to upgrade from Guest User.
* **Institutional Users:** 
  + **Super admin registers the institution** (name of institution, domain, contact details of Super Admin (name, phone no, email id, etc.), tax/gst no)
  + **Super admin invites users via email**
  + **Users complete their profile**
  + **Roll based access – Super Admin and Institutional user (Application usage is same as individual user)**

**3.2 Credit System**

* **Credit Purchase**:
  + **Individuals**: Purchase credits via payment gateway (UPI, Cards, Stripe, PayPal etc).
  + **Institutions**: Super Admin purchases bulk credits (PO/invoice support).
  + Credits displayed in dashboards (real-time updates).
* **Credit Consumption**:
  + 1 credit deducted per report generation (any button click).
  + Low credit alerts (email/in-app notifications).
  + Block report generation if credits = 0.
* **Credit Allocation for Guests:**
  + 3 free credits auto-assigned after email verification. *(Nos are flexible)*
  + Credits expire after 30 days if unused. *(Nos are flexible)*
  + Guests cannot purchase credits; must convert to Individual User.
* **Credit Consumption of Guests:**
  + Guests consume credits identically to Individual Users (1 credit/report).
  + Guests see prompts to upgrade when credits are exhausted.

**3.3 Report generation**

* Users input data via forms and or upload.
* Report generated on click.
* Unique report ID, timestamp, and download link saved in history (users’ dashboard)

**3.4 Dashboards**

* **Guest User Dashboard:**
  + Credit balance (3 free credits).
  + Report history (downloadable).
  + Prominent "Upgrade to Individual User" button for credit purchases.
  + Limited or No access to features until upgrade. *(limited or no access and features are flexible)*
* **Individual User Dashboard**:
  + Credit balance.
  + Report history (title, date, download button, report Delete Button).
  + Purchase credit button.
* **Institutional User Dashboard**:
  + Same as individual, but credits allocated by Super Admin.
* **Super Admin Dashboard**:
  + Add/remove users (bulk upload via CSV).
  + Allocate/reallocate credits to users.
  + View institutional credit balance and usage analytics.
  + Export user/report data (CSV/Excel).

**3.5 Security & Compliance**(Daily backups stored in encrypted cloud storage)

* **Data Protection**:
  + Encrypt sensitive data (AES-256).
  + Secure authentication (JWT/OAuth 2.0).
* **Compliance**:
  + GDPR, FERPA, and accessibility (WCAG 2.1).
  + Audit logs for credit transactions and user actions.
* **Guest Data:**
  + Guest accounts details remains send periodical reminders. (period is flexible)
  + Reports from guest accounts archived for 1 year.

**3.6 Notifications**

* **Email alerts for:**
  + User creation.
  + Credit purchase confirmation.
  + Low credit warnings (notify when credits are less than 5).
  + Report generation completion.
  + Unsubscribe to emails notifications.
* **Guest-specific alerts:**
  + Email confirmation with free credit details.
  + Reminders to upgrade before credit expiration.
  + Periodical reminders for inactive guest user.
  + Unsubscribe to emails notifications.

**4. Non-Functional Requirements**

* **Guest User Experience:**
  + Seamless upgrade flow from Guest to Individual user.
  + Performance optimized for high guest user traffic.
* **Performance**:
  + Support 500+ concurrent users.
  + Report generation within ≤5 seconds.
* **Scalability**:
  + Modular and flexible architecture for future feature additions.
* **UI/UX**:
  + Responsive design (desktop). Mobile version is not planned.
  + Clear navigation, accessibility features.

**5. Integration Requirements**

* **Payment Gateways**: Any Indian, UPI, Stripe, PayPal, bank transfers (for institutions).
* **Cloud Storage**: AWS S3 or Google Cloud for report storage. (or suggest)
* **Analytics**: Google Analytics or Mixpanel for usage tracking. (or suggest)

**6. Testing Requirements**

* Unit, integration, and load testing.
* User acceptance testing (UAT) with stakeholders.
* Security penetration testing.
* Validations across application, specifically guest registration, credit allocation, and upgrade workflows.
* Test credit expiration and account deletion policies.

**7. Deployment & Maintenance**

* **Hosting**: AWS/Azure with CI/CD pipeline. (or suggest)
* **Monitoring**: New Relic/Datadog for uptime and error tracking.
* **Support**: 1-year post-launch maintenance (bug fixes, updates).

**8. Deliverables**

* Fully functional web app with source code.
* Database structures scripts
* Admin and user documentation.
* Deployment scripts and test cases.

**9. Assumptions**

* Institutional users will use institutional email domains for verification.
* Reports are stored for 2 years unless deleted by users/Super Admin.

10. Note:  
Any additional features that need to be incorporated to make it a fully functional WebApp, must be included.

------------------------------------------------------------------------------------------------------------