

SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

1. Introduction

1.1 Purpose

This SRS document specifies the software requirements for the SkillSync Peer Learning Exchange Platform. It details functional and non-functional requirements for developers, testers, and stakeholders.

1.2 Scope

The system enables peer-to-peer skill exchange through:

- User profile creation and management
- AI-powered skill matching
- Session scheduling and management
- Integrated communication (chat, video)
- Rating and feedback system
- Skill credit economy
- Progress tracking and analytics

1.3 Definitions, Acronyms, and Abbreviations

- **P2P:** Peer-to-Peer
- **MAU:** Monthly Active Users
- **MVP:** Minimum Viable Product
- **API:** Application Programming Interface
- **JWT:** JSON Web Token
- **CRUD:** Create, Read, Update, Delete
- **Session:** A scheduled learning exchange between two or more users
- **Skill Credit:** Virtual currency earned by teaching, spent on learning

1.4 References

- OWASP Security Guidelines
- WCAG 2.1 Accessibility Standards
- GDPR Compliance Documentation
- REST API Best Practices

2. Overall Description

2.1 Product Perspective

The platform is a standalone web application with future mobile app support. It integrates with third-party services (video conferencing, calendar, payment processing) but maintains independent user data and core functionality.

2.2 Product Functions

1. User Authentication & Authorization
2. Profile Management
3. Skill Catalog & Taxonomy
4. Matching Algorithm Execution
5. Session Scheduling & Calendar Integration
6. Real-Time Communication
7. Video Conferencing Integration
8. Rating & Review System
9. Credit Economy Management
10. Notification System
11. Search & Discovery
12. Analytics & Reporting

2.3 User Classes and Characteristics

Learner:

- Primary goal: Acquire new skills
- Technical proficiency: Basic to intermediate
- Usage frequency: 2-4 times/week

Teacher:

- Primary goal: Share expertise, earn credits
- Technical proficiency: Intermediate to advanced
- Usage frequency: 3-5 times/week

Admin:

- Primary goal: Platform management, moderation
- Technical proficiency: Advanced
- Usage frequency: Daily

Super Admin:

- Primary goal: System configuration, analytics
- Technical proficiency: Expert
- Usage frequency: Daily

2.4 Operating Environment

- **Web Browsers:** Chrome 100+, Firefox 100+, Safari 15+, Edge 100+
- **Mobile Browsers:** iOS Safari 15+, Android Chrome 100+
- **Screen Resolutions:** 320px (mobile) to 2560px (desktop)
- **Network:** Minimum 1 Mbps for video calls

2.5 Design and Implementation Constraints

- Must comply with GDPR, CCPA data protection regulations
- Video calls rely on third-party APIs (Zoom, Google Meet)
- Payment processing through Stripe (subject to their terms)
- Must support internationalization (i18n) from architecture
- Real-time features require WebSocket support

2.6 Assumptions and Dependencies

- Users have stable internet connectivity
- Users have email addresses for authentication
- Third-party API uptime and availability
- Users consent to video/audio recording during sessions (optional)

3. Specific Requirements

3.1 Functional Requirements

3.1.1 User Management Module

FR-UM-001: User Registration

- **Description:** Users shall register using email or social OAuth (Google, LinkedIn)
- **Inputs:** Email, password (8+ chars, 1 uppercase, 1 number, 1 special) OR OAuth token
- **Processing:** Validate inputs, check for duplicate emails, hash password (bcrypt), send verification email
- **Outputs:** User account created, verification email sent, JWT token issued
- **Priority:** High

FR-UM-002: Email Verification

- **Description:** Users must verify email within 7 days of registration
- **Inputs:** Verification token from email link
- **Processing:** Validate token, mark email as verified, enable full account access
- **Outputs:** Account activated, redirect to profile setup
- **Priority:** High

FR-UM-003: User Login

- **Description:** Registered users shall login with credentials
- **Inputs:** Email and password OR OAuth
- **Processing:** Validate credentials, generate session JWT, log access
- **Outputs:** JWT token, redirect to dashboard
- **Priority:** High

FR-UM-004: Password Reset

- **Description:** Users can reset forgotten passwords
- **Inputs:** Email address
- **Processing:** Send password reset link (valid 1 hour), validate new password
- **Outputs:** Password updated, confirmation email sent
- **Priority:** Medium

FR-UM-005: Profile Creation

- **Description:** New users complete profile with skills and preferences
- **Inputs:**
 - Name (2-50 chars)
 - Profile photo (max 5MB, JPG/PNG)
 - Bio (max 500 chars)
 - Location (city, country)
 - Timezone (auto-detect with manual override)
 - 3+ teachable skills with proficiency
 - 3+ learning interests with target proficiency
 - Availability schedule (weekly calendar)
- **Processing:** Validate inputs, upload photo to CDN, store in database, calculate profile completeness
- **Outputs:** Profile saved, redirect to matching page
- **Priority:** High

FR-UM-006: Profile Editing

- **Description:** Users can update their profiles anytime

- **Inputs:** Any editable profile field
- **Processing:** Validate changes, update database, recalculate matches if skills changed
- **Outputs:** Profile updated, success message
- **Priority:** Medium

FR-UM-007: Account Deletion

- **Description:** Users can permanently delete accounts
- **Inputs:** Password confirmation
- **Processing:** Anonymize user data (GDPR compliance), cancel active sessions, refund credits
- **Outputs:** Account deleted, confirmation email
- **Priority:** Low

3.1.2 Skill Management Module

FR-SM-001: Skill Taxonomy

- **Description:** System maintains hierarchical skill categories
- **Data Structure:**
 - 12 top-level categories (Technology, Business, Creative, Language, etc.)
 - 50+ mid-level subcategories
 - 500+ specific skills
- **Processing:** Admin-managed taxonomy, searchable, filterable
- **Priority:** High

FR-SM-002: Skill Search

- **Description:** Users can search and add skills to profiles
- **Inputs:** Search query (min 2 chars)
- **Processing:** Fuzzy search across skill taxonomy, return ranked results
- **Outputs:** List of matching skills
- **Priority:** High

FR-SM-003: Custom Skill Addition

- **Description:** Users can request new skills not in taxonomy
- **Inputs:** Skill name, category suggestion, description
- **Processing:** Admin review queue, approve/reject, add to taxonomy if approved
- **Outputs:** Skill added to user profile (pending or approved)
- **Priority:** Medium

FR-SM-004: Skill Proficiency Levels

- **Description:** Each skill has 4 proficiency levels
- **Levels:** Beginner (0-25%), Intermediate (26-60%), Advanced (61-85%), Expert (86-100%)
- **Processing:** User self-assessment + peer validation
- **Priority:** Medium

3.1.3 Matching Algorithm Module

FR-MA-001: Match Generation

- **Description:** System generates personalized matches daily
- **Inputs:** User profile (skills, preferences, availability, ratings, activity)
- **Processing:**
 - Calculate complementary skill scores
 - Factor in timezone compatibility
 - Consider rating thresholds (minimum 3.5/5)
 - Apply content-based + collaborative filtering
 - Rank by composite match score (0-100)
- **Outputs:** Top 20 match suggestions per user
- **Priority:** High
- **Algorithm Weights:**
 - Skill complementarity: 40%
 - Availability overlap: 20%
 - Past success rate: 15%
 - Rating compatibility: 15%
 - Response rate: 10%

FR-MA-002: Match Explanation

- **Description:** Each match includes reasoning
- **Outputs:** "90% match because: You both want to exchange Design ↔ Coding, 15+ overlapping hours, both highly rated (4.8+)"
- **Priority:** Medium

FR-MA-003: Match Filtering

- **Description:** Users filter matches by criteria
- **Filters:** Skill category, proficiency level, location, availability, rating (min threshold)
- **Processing:** Client-side filtering on initial 20 matches, server-side if expanded
- **Priority:** Medium

FR-MA-004: Match Actions

- **Description:** Users can favorite, pass, or block matches
- **Inputs:** Match ID, action type
- **Processing:** Update user preferences, exclude blocked users from future matches
- **Outputs:** Match list updated, algorithm learns preferences
- **Priority:** Medium

3.1.4 Communication Module

FR-CM-001: Real-Time Chat

- **Description:** Matched users communicate via text chat
- **Inputs:** Message text (max 1000 chars)
- **Processing:** WebSocket transmission, encryption in transit, spam filtering, store in database
- **Outputs:** Message delivered, read receipts, typing indicators
- **Priority:** High
- **Technical:** Socket.io with Redis pub/sub for scaling

FR-CM-002: File Sharing

- **Description:** Users share resources within chat
- **Inputs:** File upload (max 10MB, allowed: PDF, DOC, JPG, PNG, MP4, ZIP)
- **Processing:** Virus scan, upload to S3, generate secure CDN URL, store metadata
- **Outputs:** File available for download
- **Priority:** Medium

FR-CM-003: Message Search

- **Description:** Users search chat history
- **Inputs:** Search query
- **Processing:** Full-text search across messages with that user
- **Outputs:** List of matching messages with context
- **Priority:** Low

FR-CM-004: Report/Block Users

- **Description:** Users report inappropriate behavior or block users
- **Inputs:** User ID, report reason, optional description
- **Processing:** Create moderation ticket, immediately block if selected, notify admin team
- **Outputs:** User blocked from contacting reporter, admin review initiated
- **Priority:** High

3.1.5 Session Management Module

FR-SSM-001: Session Proposal

- **Description:** Users propose session dates/times
- **Inputs:**
 - Recipient user ID
 - Session type (one-time, recurring)
 - Duration (30min, 1hr, 2hr, custom)
 - Skill to exchange
 - Up to 5 proposed time slots
 - Optional agenda/notes
- **Processing:** Check recipient availability, send notification, create pending session
- **Outputs:** Session proposal sent, awaiting confirmation
- **Priority:** High

FR-SSM-002: Session Response

- **Description:** Recipient can accept, decline, or counter-propose
- **Inputs:** Session ID, action (accept/decline/counter)
- **Processing:** If accepted, create confirmed session, generate video link, send calendar invites; if counter-proposed, create new proposal
- **Outputs:** Session confirmed/declined, both parties notified
- **Priority:** High

FR-SSM-003: Calendar Integration

- **Description:** Confirmed sessions sync with Google/Outlook calendars
- **Inputs:** User OAuth tokens for calendar access
- **Processing:** Create calendar event with video link, set reminders (24hr, 1hr, 15min before)
- **Outputs:** Calendar event created, confirmation shown
- **Priority:** Medium

FR-SSM-004: Video Link Generation

- **Description:** Each session gets a unique video meeting link
- **Processing:** Call Zoom API / [Daily.co](https://daily.co) API to create room, embed link in session details
- **Outputs:** Meeting link available 15 minutes before session
- **Priority:** High

FR-SSM-005: Session Reminders

- **Description:** Automated reminders sent via email and push

- **Schedule:** 24 hours, 1 hour, 15 minutes before session
- **Processing:** Scheduled jobs check upcoming sessions, trigger notifications
- **Priority:** Medium

FR-SSM-006: Session Rescheduling

- **Description:** Either party can request reschedule
- **Inputs:** New proposed times
- **Processing:** Send reschedule request, await confirmation, update calendar if accepted
- **Outputs:** Session rescheduled or original time retained
- **Priority:** Medium

FR-SSM-007: Session Cancellation

- **Description:** Either party can cancel with reason
- **Inputs:** Session ID, cancellation reason
- **Processing:** Refund credits if within policy (24hr+ notice), notify both parties, remove from calendars
- **Outputs:** Session cancelled, credits handled
- **Priority:** Medium

3.1.6 Rating & Feedback Module

FR-RF-001: Post-Session Rating

- **Description:** Both parties rate session within 48 hours
- **Inputs:**
 - Overall rating (1-5 stars)
 - Dimension ratings:
 - Knowledge/Skill Transfer (1-5)
 - Communication Quality (1-5)
 - Professionalism (1-5)
 - Written review (optional, max 500 chars)
 - Tags (punctual, patient, knowledgeable, etc.)
- **Processing:** Store ratings, calculate weighted averages, update user profiles, send to recipient
- **Outputs:** Rating submitted, impacts user's aggregate score
- **Priority:** High

FR-RF-002: Rating Enforcement

- **Description:** Users must rate before booking next session

- **Processing:** Check for pending ratings on profile/booking pages
- **Outputs:** Block new bookings if ratings overdue (48hrs+)
- **Priority:** Medium

FR-RF-003: Rating Display

- **Description:** Aggregate ratings shown on profiles
- **Display:**
 - Overall average (weighted by recency: last 3 months 60%, 3-12 months 40%)
 - Total sessions completed
 - Individual dimension averages
 - Recent reviews (last 5)
- **Priority:** High

FR-RF-004: Rating Disputes

- **Description:** Users can dispute unfair ratings
- **Inputs:** Rating ID, dispute reason
- **Processing:** Admin review, may remove rating if violates policy
- **Outputs:** Rating removed or dispute rejected
- **Priority:** Low

3.1.7 Credit Economy Module

FR-CE-001: Credit Allocation

- **Description:** New users receive starter credits
- **Amount:** 50 credits upon profile completion
- **Priority:** High

FR-CE-002: Earning Credits

- **Description:** Users earn credits by teaching
- **Rules:**
 - 10 credits per completed 1-hour teaching session
 - Prorated for other durations (5 credits for 30min, 20 for 2hr)
 - Credits awarded after both parties submit ratings
- **Processing:** Calculate credits, add to balance, notify user
- **Priority:** High

FR-CE-003: Spending Credits

- **Description:** Users spend credits to book sessions

- **Rules:**
 - Standard sessions: 10 credits per hour
 - Premium teachers (4.8+ rating, 50+ sessions): 15-20 credits per hour
 - Group sessions: 5 credits per hour per person
- **Processing:** Deduct credits on session confirmation, hold in escrow until completion
- **Priority:** High

FR-CE-004: Credit Purchase

- **Description:** Users can buy credits with real money
- **Packages:**
 - 100 credits: \$10 (10¢/credit)
 - 500 credits: \$45 (9¢/credit, 10% bonus)
 - 1000 credits: \$80 (8¢/credit, 20% bonus)
- **Processing:** Stripe payment integration, add credits on successful payment
- **Priority:** Medium

FR-CE-005: Credit Refunds

- **Description:** Credits refunded for cancelled sessions (conditions apply)
- **Rules:**
 - Full refund: cancelled 24+ hours before
 - 50% refund: cancelled 2-24 hours before
 - No refund: cancelled <2 hours before (unless teacher cancels)
- **Processing:** Return credits to spender, notify both parties
- **Priority:** Medium

FR-CE-006: Credit Expiration

- **Description:** Credits expire after 12 months of account inactivity
- **Processing:** Scheduled job checks last login, expires old credits, sends warning emails at 11 months
- **Priority:** Low

FR-CE-007: Credit History

- **Description:** Users view transaction history
- **Display:** Date, type (earned/spent/purchased/refunded), amount, balance, related session
- **Priority:** Low

3.1.8 Notification Module

FR-NM-001: Notification Types

- **Categories:**
 - Session-related (proposals, confirmations, reminders, cancellations)
 - Matching (new match suggestions)
 - Messaging (new chat messages)
 - Credits (earned, spent, low balance warnings)
 - System (profile verification, policy updates)
- **Priority:** High

FR-NM-002: Notification Channels

- **Channels:**
 - In-app notifications (bell icon with badge count)
 - Email notifications (customizable per category)
 - Push notifications (mobile web/future apps)
- **Priority:** High

FR-NM-003: Notification Preferences

- **Description:** Users customize notification settings
- **Options:** Enable/disable per category per channel
- **Processing:** Store preferences, respect settings when sending notifications
- **Priority:** Medium

FR-NM-004: Notification History

- **Description:** Users view past notifications
- **Display:** Last 30 days, mark as read/unread, delete
- **Priority:** Low

3.1.9 Search & Discovery Module

FR-SD-001: User Search

- **Description:** Search for users by name, skills, location
- **Inputs:** Search query, filters (skill, location, rating, availability)
- **Processing:** Full-text search with ranking, filter results
- **Outputs:** Paginated user results (20 per page)
- **Priority:** Medium

FR-SD-002: Skill-Based Browse

- **Description:** Browse users by specific skills
- **Inputs:** Skill category or specific skill
- **Processing:** Query users with that skill, sort by rating/recency
- **Outputs:** List of users teaching that skill
- **Priority:** Medium

FR-SD-003: Featured Teachers

- **Description:** Highlight top-rated, active teachers
- **Criteria:** 4.5+ rating, 20+ sessions, active within 7 days
- **Display:** Rotating carousel on homepage/dashboard
- **Priority:** Low

3.1.10 Admin Module

FR-AM-001: User Management

- **Description:** Admins view/edit/suspend/delete user accounts
- **Actions:** View full profile, edit details, suspend (temp), ban (permanent), delete
- **Priority:** Medium

FR-AM-002: Content Moderation

- **Description:** Review reported users/messages
- **Workflow:** Review queue, view evidence, take action (warn, suspend, ban), notify reporter
- **Priority:** High

FR-AM-003: Skill Taxonomy Management

- **Description:** Add/edit/remove skills from catalog
- **Actions:** CRUD operations on skills, approve user-submitted skills
- **Priority:** Medium

FR-AM-004: Analytics Dashboard

- **Description:** View platform metrics
- **Metrics:**
 - Total users, MAU, new registrations (daily/weekly/monthly)
 - Total sessions, completion rate, average ratings
 - Credit economy (earned, spent, purchased)
 - Top skills, top teachers, user retention
- **Priority:** Medium

FR-AM-005: System Configuration

- **Description:** Super admins configure platform settings
- **Settings:** Credit pricing, session durations, refund policies, matching algorithm weights
- **Priority:** Low

3.2 Non-Functional Requirements

3.2.1 Performance Requirements

NFR-P-001: Response Time

- Web pages shall load in < 2 seconds on 4G connection
- API responses shall complete in < 200ms (p95)
- Search queries shall return results in < 500ms
- Matching algorithm shall complete in < 3 seconds

NFR-P-002: Throughput

- System shall support 10,000 concurrent users
- Chat system shall handle 1,000 messages/second
- Video integration shall support 500 concurrent sessions

NFR-P-003: Scalability

- Horizontal scaling via load balancers
- Database sharding for user data
- CDN for static assets and media
- Microservices architecture for independent scaling

3.2.2 Security Requirements

NFR-S-001: Authentication

- Multi-factor authentication (optional, recommended)
- JWT tokens with 7-day expiration
- Refresh tokens with 30-day expiration
- OAuth 2.0 for social login

NFR-S-002: Authorization

- Role-based access control (User, Teacher, Admin, Super Admin)
- API endpoints protected with middleware
- Users can only access their own data (except public profiles)

NFR-S-003: Data Protection

- All traffic over HTTPS/TLS 1.3
- Passwords hashed with bcrypt (cost factor 12)
- Sensitive data encrypted at rest (AES-256)
- PII anonymized in analytics

NFR-S-004: Compliance

- GDPR: right to access, rectify, delete data
- CCPA: opt-out of data sales (not applicable, but supported)
- Data retention: user data deleted 30 days after account deletion

NFR-S-005: Vulnerability Management

- Regular penetration testing (quarterly)
- Dependency vulnerability scanning (automated)
- OWASP Top 10 mitigation
- Rate limiting on authentication endpoints (5 attempts per minute)

3.2.3 Usability Requirements

NFR-U-001: Accessibility

- WCAG 2.1 Level AA compliance
- Screen reader support (ARIA labels)
- Keyboard navigation for all features
- Minimum color contrast ratio 4.5:1

NFR-U-002: Responsiveness

- Mobile-first design
- Support screen widths 320px - 2560px
- Touch-optimized UI elements (44×44px minimum)
- Responsive images and lazy loading

NFR-U-003: Internationalization

- Support UTF-8 character encoding
- Date/time formatting per user locale
- Prepared for multi-language support (English initially)

NFR-U-004: Browser Compatibility

- Chrome 100+ (priority 1)
- Firefox 100+ (priority 1)
- Safari 15+ (priority 2)

- Edge 100+ (priority 2)
- IE 11 (not supported)

3.2.4 Reliability Requirements

NFR-R-001: Availability

- 99.9% uptime (< 8.76 hours downtime/year)
- Planned maintenance during low-traffic hours (2-4 AM UTC)
- Maximum 4 hours planned downtime per month

NFR-R-002: Fault Tolerance

- Graceful degradation if third-party services fail
- Circuit breakers for external API calls
- Fallback messaging if video fails
- Retry logic for failed transactions

NFR-R-003: Backup & Recovery

- Automated daily backups (retained 30 days)
- Point-in-time recovery capability
- RTO (Recovery Time Objective): 4 hours
- RPO (Recovery Point Objective): 1 hour

3.2.5 Maintainability Requirements

NFR-M-001: Code Quality

- 80%+ unit test coverage
- Automated CI/CD pipeline
- Code review required for all merges
- Linting and formatting enforced (ESLint, Prettier)

NFR-M-002: Documentation

- API documentation (OpenAPI/Swagger)
- System architecture diagrams
- Database schema documentation
- Deployment runbooks

NFR-M-003: Monitoring

- Application performance monitoring (APM)

- Error tracking and alerting
- Logging (centralized via ELK stack or similar)
- Uptime monitoring (external service)

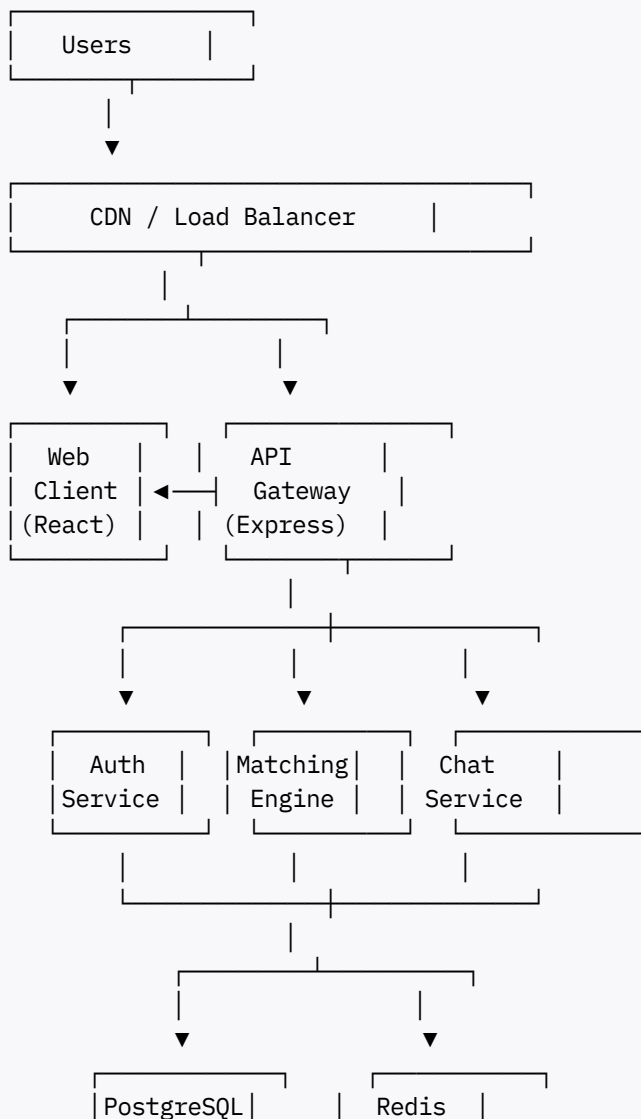
3.2.6 Portability Requirements

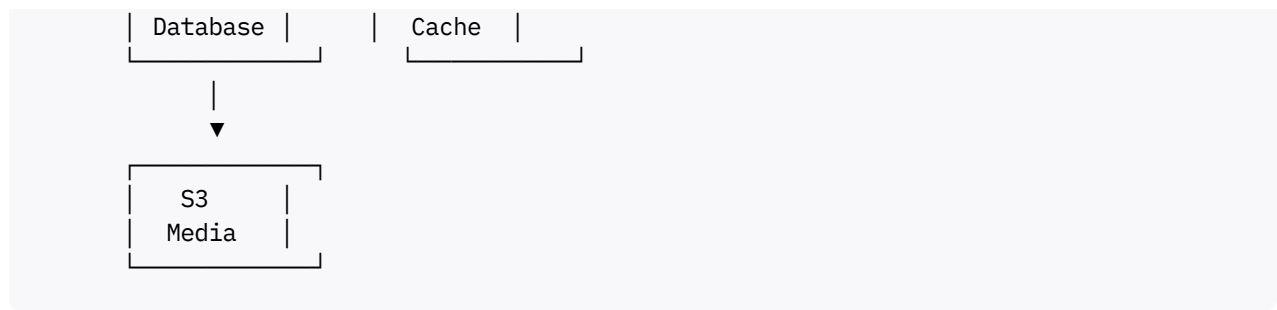
NFR-PO-001: Platform Independence

- Containerized deployment (Docker)
- Cloud-agnostic architecture (AWS initially, but portable)
- Database migrations managed (e.g., Prisma, TypeORM)

4. System Architecture

4.1 High-Level Architecture





4.2 Database Schema (Key Tables)

users

- id (PK), email, password_hash, name, profile_photo_url, bio, location, timezone, verification_status, created_at, updated_at

skills

- id (PK), name, category, subcategory, description

user_skills

- id (PK), user_id (FK), skill_id (FK), skill_type (teach/learn), proficiency_level, verified

matches

- id (PK), user1_id (FK), user2_id (FK), match_score, status (suggested/favorited/passed/blocked), created_at

sessions

- id (PK), requester_id (FK), recipient_id (FK), skill_id (FK), scheduled_at, duration_minutes, status (proposed/confirmed/completed/cancelled), video_link, credits_cost, created_at, updated_at

messages

- id (PK), sender_id (FK), recipient_id (FK), message_text, file_url, read_at, created_at

ratings

- id (PK), session_id (FK), rater_id (FK), ratee_id (FK), overall_rating, knowledge_rating, communication_rating, professionalism_rating, review_text, created_at

credits

- id (PK), user_id (FK), amount, balance, transaction_type (earned/spent/purchased/refunded), related_session_id, created_at

notifications

- id (PK), user_id (FK), type, title, message, link, read_at, created_at

5. External Interface Requirements

5.1 User Interface

UI-001: Login/Signup Page

- Clean, minimal design
- Email/password fields with validation
- Social login buttons (Google, LinkedIn)
- "Forgot Password" link

UI-002: Dashboard

- Welcome message with profile completeness
- Quick stats (credits, sessions completed, current rating)
- Top 5 match suggestions cards
- Upcoming sessions list
- Recent notifications

UI-003: Match Discovery Page

- Filter sidebar (skills, location, availability, rating)
- Match cards with photo, name, skills, match score, "Connect" button
- Infinite scroll or pagination

UI-004: Profile Page

- Editable sections: photo, bio, skills, availability
- Display: rating, sessions completed, reviews
- Verification badges

UI-005: Session Management Page

- Tabs: Upcoming, Past, Proposals
- Session cards with details, video link, actions (reschedule, cancel, rate)

UI-006: Chat Interface

- List of conversations (left sidebar)
- Message thread (right panel)
- File attachment button
- User info panel

5.2 Hardware Interfaces

- No direct hardware interfaces
- Client devices: Desktops, laptops, tablets, smartphones
- Server hardware: Cloud-based (AWS EC2, RDS, etc.)

5.3 Software Interfaces

SI-001: Video Conferencing API

- Provider: Zoom or [Daily.co](#)
- Purpose: Generate meeting links, embed video
- Interface: REST API, webhooks for recording/events

SI-002: Calendar API

- Provider: Google Calendar API, Microsoft Graph
- Purpose: Sync sessions to user calendars
- Interface: OAuth 2.0, REST API

SI-003: Payment Gateway

- Provider: Stripe
- Purpose: Process credit purchases
- Interface: Stripe.js SDK, webhooks

SI-004: Email Service

- Provider: SendGrid or AWS SES
- Purpose: Transactional emails (verification, notifications)
- Interface: REST API

SI-005: Cloud Storage

- Provider: AWS S3 or Cloudflare R2
- Purpose: Store profile photos, shared files
- Interface: SDK (boto3, AWS SDK)

SI-006: Push Notification Service

- Provider: Firebase Cloud Messaging
- Purpose: Browser/mobile push notifications
- Interface: FCM SDK

5.4 Communication Interfaces

CI-001: HTTP/HTTPS

- All web traffic over HTTPS
- REST API follows JSON format
- Status codes: 200 (success), 400 (client error), 401 (unauthorized), 404 (not found), 500 (server error)

CI-002: WebSocket

- Real-time chat and notifications
- Protocol: Socket.io over WebSocket
- Fallback to long-polling if WebSocket unavailable

CI-003: Webhooks

- Receive events from Stripe (payments), video platforms (recordings)
- Authenticated via signature verification

6. Testing Requirements

6.1 Unit Testing

- All business logic functions
- Target: 80%+ code coverage
- Framework: Jest (JavaScript)

6.2 Integration Testing

- API endpoint testing
- Database operations
- Third-party service integrations
- Framework: Supertest, Postman

6.3 End-to-End Testing

- Critical user flows:
 - Complete signup to first session
 - Session booking and completion
 - Credit purchase and usage
- Framework: Cypress or Playwright

6.4 Performance Testing

- Load testing: 10,000 concurrent users
- Stress testing: find breaking point
- Tools: Apache JMeter, K6

6.5 Security Testing

- Penetration testing (manual + automated)
- Vulnerability scanning: OWASP ZAP
- SQL injection, XSS, CSRF prevention validation

6.6 Usability Testing

- User testing with 10-20 beta users
- Task completion rate > 90%
- System Usability Scale (SUS) score > 75

6.7 Acceptance Testing

- Validate all functional requirements
- Business stakeholder sign-off
- Acceptance criteria from PRD verified

7. Appendices

Appendix A: Glossary

- **Session:** A scheduled skill exchange meeting
- **Credit:** Platform currency for bookings
- **Match Score:** Algorithm-generated compatibility rating (0-100)
- **Skill Proficiency:** User's expertise level in a skill

Appendix B: Revision History

Version	Date	Author	Changes
1.0	Oct 25, 2025	[Your Name]	Initial SRS creation

Next Steps & Recommendations

Immediate Actions:

1. **Validate with Users:** Conduct 10-15 user interviews with target personas to validate problem assumptions and feature priorities
2. **Competitive Analysis:** Deep dive into existing platforms (SkillSwap, Trinsic mentioned earlier) to identify their weaknesses
3. **Technical Feasibility:** Prototype the AI matching algorithm with sample data to validate approach
4. **Design Mockups:** Create high-fidelity UI mockups for core flows (onboarding, matching, booking)
5. **Development Planning:** Break down into 2-week sprints, starting with authentication and profile creation

Differentiation Summary:

Your platform's **unique competitive advantages:**

1. **Multi-dimensional AI matching** (beyond simple skill complementarity)
2. **Skill credit economy with gamification** (not just free or fully paid)
3. **Trust layer** (skill verification, progressive trust levels)
4. **Micro-learning integration** (15-min quick sessions for busy users)
5. **Community learning circles** (group cohorts, not just 1:1)
6. **Future-ready** (VR/AR roadmap for immersive learning)

This positions you between traditional learning platforms (Coursera, Udemy) and informal skill swaps, offering structured peer learning with built-in quality assurance.

Would you like me to create the UI wireframes, develop the matching algorithm prototype, or dive deeper into any specific section of the PRD/SRS?



1. <https://www.paradisosolutions.com/blog/best-peer-to-peer-learning-platforms/>
2. <https://sanalabs.com/learn-blog/ai-learning-platforms-2025>
3. <https://www.educate-me.co/blog/best-peer-to-peer-learning-platforms>
4. <https://www.digitallearninginstitute.com/blog/education-technology-trends-to-watch-in-2025>
5. <https://www.linkedin.com/pulse/on-demand-skill-exchange-platform-mohamed-yasser-ednqc>
6. https://www.ijasret.com/VolumeArticles/FullTextPDF/1478_Skill_Share_A_Knowledge_Sharing_Platform.pdf
7. <https://elearningindustry.com/elearning-in-2025-key-trends-shaping-the-future-of-education>
8. <https://www.articulate.com/blog/2025-e-learning-trends-whats-in-and-whats-out/>
9. <https://www.devlinpeck.com/content/elearning-trends>

10. https://ijirt.org/publishedpaper/IJIRT177304_PAPER.pdf
11. <https://sparkco.ai/blog/enhancing-learning-a-guide-to-peer-learning-programs>
12. <https://novoresume.com/career-blog/learning-and-development-trends>
13. <https://www.zigpoll.com/content/how-do-users-perceive-and-interact-with-the-collaborative-features-on-peertopeer-learning-platforms-and-what-design-elements-most-effectively-enhance-their-engagement-and-knowledge-sharing>
14. <https://whatfix.com/blog/adaptive-learning-platforms/>
15. <https://www.disco.co/blog/top-10-ai-e-learning-platforms-2025>
16. <https://www.shrm.org/topics-tools/news/organizational-employee-development/peer-to-peer-learning-evolves>
17. <https://www.absorblms.com/blog/best-learning-experience-platforms/>
18. <https://business.adobe.com/summit/2025/sessions/skill-exchange-managing-the-unexpected-with-rtcdp-s902.html>
19. <https://fossunited.org/hack/fosshack25/p/rbbd7rpub8>
20. <https://360learning.com/blog/ai-learning-platforms/>