Great! Here's a comprehensive **Page-by-Page Functional Requirements Table** for your freelancing platform website (similar to Fiverr/Upwork). This will help you track **UI pages**, **features**, **form validations**, **actions**, and **data interactions** for every page in your application.

**✅ Functional Requirements Table (Page by Page)**

| **#** | **Page Name** | **User Type** | **Features / Functionalities** | **Validations / Constraints** | **Backend Actions / DB Interaction** |
| --- | --- | --- | --- | --- | --- |
| 1 | **Landing Page** | All | - View intro, how it works, features, testimonials, CTA buttons- Option to login/register | N/A | None |
| 2 | **Login Page** | All | - Login form with email & password- Redirect based on role (Freelancer/Client/Admin) | - Email must be valid- Password min 6 chars- Show error on invalid creds | - Authenticate user- Generate JWT- Save session or cookie |
| 3 | **Register Page** | Freelancer/Client | - Registration form with name, email, password, confirm password, user type selector | - All fields mandatory- Password = Confirm password- Email unique- Strong password required | - Save user to DB- Check email uniqueness- Hash password |
| 4 | **Client Dashboard** | Client | - View posted projects- Post new project- View bids on projects- View freelancer profiles- Chat option | - Auth required- Pagination on list | - Fetch projects by client- Save project to DB- Fetch bids |
| 5 | **Post Project Page** | Client | - Title, Description, Skills required, Budget, Deadline- Submit project | - All fields required- Budget must be numeric- Date in future | - Insert project in DB with client ID |
| 6 | **Freelancer Dashboard** | Freelancer | - View available projects- Search/filter projects- View applied bids- Chat option | - Auth required | - Fetch all open projects- Fetch own bids |
| 7 | **Project Details Page** | All | - Project title, description, client name, budget, apply button (freelancer), bid list (client) | - Must be logged in | - Fetch full project detail- Fetch bids by project ID |
| 8 | **Apply/Bid Page** | Freelancer | - Bid Amount, Proposal text, Duration- Submit bid | - All fields required- Bid must be numeric and within project budget | - Insert bid into DB with projectId and freelancerId |
| 9 | **Profile Page (Client)** | Client | - Name, Company Name, Profile Picture, Posted Projects, Ratings from freelancers | - Image size limit- Bio char limit | - Fetch profile details from DB- Update on edit |
| 10 | **Profile Page (Freelancer)** | Freelancer | - Name, Skills, Bio, Projects Done, Portfolio Upload, Ratings, Education | - Mandatory: Skills, Bio- Portfolio = PDF/JPG/PNG- Image size max 5MB | - Fetch and update profile- Upload documents/images to server |
| 11 | **Search Freelancers** | Client | - Search by skill, rating, location- View mini-profiles- Filter/sort | - Input sanitized | - Fetch filtered freelancer list from DB |
| 12 | **Search Projects** | Freelancer | - Search by title, budget range, category, client rating | - Input sanitized | - Fetch filtered project list from DB |
| 13 | **Chat Page** | Client/Freelancer | - One-on-one messaging- Real-time updates (WebSocket)- Chat history | - Authenticated access only | - Fetch chat history- Save new messages- Use WebSocket or polling |
| 14 | **Contract/Offer Page** | Client | - Send offer to freelancer- Set terms, milestones, payment agreement | - Validate amount and dates | - Save contract to DB- Notify freelancer |
| 15 | **Contract Review Page** | Freelancer | - Accept/reject offer- View milestones, timelines | - Reject only if not started- Accept = binds contract | - Update contract status in DB |
| 16 | **Payment Page** | Client | - Pay milestone, Full payment- View transaction history- Secure payment integration (Stripe/Razorpay) | - Payment amount validation- Auth check- Retry on failure | - Save payment transaction- Update project/payment status |
| 17 | **Ratings & Reviews** | Client/Freelancer | - Leave feedback after project- 1–5 rating, comments | - Only after project completion- Max 500 char comment | - Save rating/review to DB- Update freelancer/client avg rating |
| 18 | **Admin Dashboard** | Admin | - View/manage users, projects, disputes, payments, reports | - Auth with admin role- All actions logged | - Fetch all data- Enable delete/block users/projects |
| 19 | **Notifications Page** | All | - Show new offers, messages, status changes- Mark as read | - Auth check | - Fetch notifications from DB- Mark as read |
| 20 | **Dispute Resolution** | Admin/Freelancer/Client | - Raise dispute- Admin review- Resolution result | - Only for valid contract- Allow one dispute at a time | - Insert dispute record- Update dispute status |
| 21 | **Settings Page** | All | - Change password, email- Enable 2FA- Deactivate account | - Password strength check- Email uniqueness- 2FA setup validation | - Update user profile/settings- Deactivate flag in DB |
| 22 | **Logout** | All | - End session and redirect to login | N/A | - Clear session/JWT/cookie |
| 23 | **Error Pages** | All | - 404 Not Found, 401 Unauthorized, 500 Server Error | - Friendly error message | - Render fallback component |

Would you like this table in **Excel or PDF** format? I can prepare that too.  
Let me know if you want to expand the features (like adding **referrals**, **wallet system**, **blog**, etc.).