My Pages Module – Full Technical, UX, API, and RAG Ingestion Documentation

Runtime URL: `https://connecwrk.com/my-pages/`

---

1. Purpose and Role in the Platform

My Pages is the user's personal hub for everything they *own* or *have published* on ConnecWrk. Instead of forcing users to jump between MSME dashboards, Talent dashboards, Freelancer dashboards, and Job Posts, this module centralizes:

- Business Pages the user created (MSME pages)
- Talent Profiles the user created (speaker/artist/trainer pages)
- Jobs the user posted
- Freelancer Profile (their own service-provider identity)

In other words: **This is "my published presence on the platform" in one screen.**

This page is also a key surface for: - onboarding: exposes missing assets (“You have not created a ___ yet”) - retention: reminds the user of what they've built so far - upgrade intent: shows gaps that can later feed premium upsell, credibility badges, etc.

---

2. High-Level UX / Layout Walkthrough

2.1 Global Layout

From the screenshot: - **Left sidebar** (standard ConnecWrk sidebar): - User mini-card (avatar, name, profile completion %) - Nudge CTA (e.g. "If you are a Talent/Artist Create your profile and connect with clients") - Quick stats: connections count, profile views - Navigation shortcuts (MSME Connect, Freelancer Connect, Assignments, Talent Connect, MSME Jobs, Articles & Trivia)

- **Top bar / header bar**:
- Global search input (placeholder like "Search 'MSME Job'")

- Global nav icons: Home, Contacts, Messages, Notifications, Profile avatar

- **Main content area**:

- Card titled **"My Pages"**
- Tab-like strip of internal anchors:
    - `Business Pages`
    - `Talent Profiles`
    - `Jobs Posted`
    - `Freelancer Profile`

- Then four stacked sections with gray headers & table-row style bodies:
    1. **My Business Pages**
    2. **Talent Profiles Created By Me**
    3. **Jobs Posted By Me**
    4. **My Freelancer Profile**

Each section either lists content (cards/rows) OR shows an empty state message:

> "You have not created a MSME Page yet"

The UX intentionally mirrors the mental model of "one person can wear multiple hats": - *MSME owner - Talent/Performer/Trainer - Recruiter / hiring manager - Freelancer looking for gigs*

The tabs at the top are **internal scroll anchors**, not separate routes. Clicking `Talent Profiles` scrolls the viewport down to the `Talent Profiles Created By Me` block, etc.

---

3. Data Surfaces / Sections Breakdown

3.1 My Business Pages

Represents MSME pages (a.k.a. business / company profiles) owned by the logged-in user.

Each record typically shows: - Business / company name - (Logo or banner image) - Industry / category - Possibly location - Possibly verification / trust badges (for premium tiers) - CTA link to open that MSME profile page for full edit/view

Empty state copy (example):

> "You have not created a MSME Page yet"

This is also the natural funnel entry for "Create Business Page" (MSME onboarding).

3.2 Talent Profiles Created By Me

Represents Talent / Artist / Speaker / Trainer profiles the user controls.

Each record typically shows: - Talent display name (stage name or professional name) - Category / specialization (e.g. "Motivational Speaker", "MC / Host", "Performer") - Profile photo/avatar - Short description / pitch line - CTA to view the public talent profile page (or edit it)

Empty state copy (example):

> "You have not created a Talents Profile yet"

This points users toward Talent Connect onboarding.

3.3 Jobs Posted By Me

Represents job listings (open roles, gigs, assignments) the user has posted.

Each record (when present) normally includes: - Job title / role name - Location (city / remote flag) - Timestamp (e.g. "Posted 2d ago") - Short job description / requirements preview - Possibly status (Open / Closed) - CTA such as "Manage Job" or "View Job"

Empty state copy (example):

> "You have not posted any jobs yet"

This is the natural funnel toward "Post Assignment" / "Post Job" flows.

3.4 My Freelancer Profile

Represents the user's *own* Freelancer persona (skills for hire).

Each record (if exists) usually includes: - Freelancer name (their own name / brand) - Headline ("UI/UX Designer", "Full Stack Developer", etc.) - Key skills / categories - Experience level or years - Possibly portfolio / rate visibility - CTA to open the full freelancer profile

Empty state copy (example):

> "You have not created a Freelancer Profile yet"

This funnels into "Create Freelancer Page" onboarding.

---

4. API Surface / Data Contracts

> All endpoints require **Bearer token auth**. Token is taken from localStorage and attached in the `Authorization: Bearer <token>` header. Pagination for lists is page-based, `page=1` for initial.

4.1 Fetch My MSME / Business Pages

Endpoint: `GET /api/msme/msmelisting`

Query params (for this module context): - `page=1` - `type=self` → tells backend "only pages owned by logged-in user"

Purpose: hydrate "My Business Pages" section.

Response (representative):

```
{
  "success": true,
  "data": [
    {
      "msmeId": "msme_123",
      "companyBusinessName": "Acme Design Studio",
      "industry": "Creative Services",
      "logo": "acme-logo.jpg",
      "location": "Mumbai",
      "isVerified": true
    }
  ],
  "currentPage": 1,
  "totalPages": 1
}
```

Notes: - If `data` is empty → show empty state. - `isVerified` can later power premium/credibility badges.

4.2 Fetch My Talent Profiles

Endpoint: `GET /api/talent/mytalentlist`

Query params: - `page=1`

Purpose: hydrate "Talent Profiles Created By Me".

Response (representative):

```
{
  "success": true,
  "result": [
    {
      "talentId": "tal_987",
      "talentName": "Rhea Kapoor",
      "categoryName": "Motivational Speaker",
      "profilePhoto": "rhea.jpg",
      "shortDescription": "Leadership talks for growth-stage teams"
    }
  ],
  "currentPage": 1,
  "totalPages": 1
}
```

Notes: - Use `talentId` to deep-link to Talent Profile Detail page.

4.3 Fetch My Job Posts

Endpoint: `GET /api/job/myjoblist`

Query params: - `page=1`

Purpose: hydrate "Jobs Posted By Me".

Response (representative):

```
{
  "success": true,
  "jobs": [
    {
      "jobId": "job_456",
      "title": "Content Strategist (Remote)",
      "location": "Remote / Bangalore Preferred",
      "postedAt": "2025-10-20T09:30:00Z",
      "description": "Own blog strategy, social campaigns, and long-form copy...",
      "status": "Open"
    }
  ],
  "currentPage": 1,
  "totalPages": 3
}
```

Notes: - `status` can be used to show badges (Open / Closed / Draft). - `postedAt` is rendered as "x hours ago" / "x days ago" in UI copy.

4.4 Fetch My Freelancer Profile(s)

Endpoint: `GET /api/project/myfreelancerlist`

Query params: - `page=1`

Purpose: hydrate "My Freelancer Profile".

Response (representative):

```
{
  "success": true,
  "data": [
    {
      "freelancerId": "free_222",
      "fullName": "Ishan Mehta",
      "headline": "Full Stack Developer / React & Node.js",
      "skills": ["React", "Node.js", "REST APIs", "AWS"],
      "yearsExperience": 4,
      "profilePhoto": "ishan.jpg",
      "bio": "I help startups build production-ready web apps fast."
    }
  ],
  "currentPage": 1,
  "totalPages": 1
}
```

Notes: - If multiple freelancer personas are allowed later (e.g. "Designer" + "Developer"), this array can have >1 entry.

---

5. Component Logic / Frontend Behavior

5.1 Initial Load

1. On mount:
2. Read bearer token from localStorage.
3. Fire all four fetches in parallel:
   - `/api/msme/msmelisting?page=1&type=self`
   - `/api/talent/mytalentlist?page=1`
   - `/api/job/myjoblist?page=1`
   - `/api/project/myfreelancerlist?page=1`
4. Store each result slice in component state.
5. Render UI sections using those slices.
6. If any slice is empty → show the corresponding empty state string.

5.2 Tab Strip / Anchor Scroll

- The top strip buttons aren't actual route changes.
- Clicking `Talent Profiles` scrolls smoothly to `Talent Profiles Created By Me`, etc.
- Implementation detail: scrollIntoView with an offset OR window.scrollTo({ top: sectionOffset - headerHeight, behavior: 'smooth' }).

5.3 Empty States

- Copy MUST be specific:
- MSME: "You have not created a MSME Page yet"
- Talent: "You have not created a Talents Profile yet"
- Jobs: "You have not posted any jobs yet"

- Freelancer: "You have not created a Freelancer Profile yet"

- These strings are important for:

- **Growth prompts:** They act like calls-to-action.
- **RAG ingestion:** They signal "user has gap X" (see RAG section below).

5.4 Access / Edit CTAs

Not shown in the provided screenshot, but expected behavior: - Clicking a card row (MSME / Talent / Job / Freelancer) should deep-link to the appropriate management or public page, e.g.: - `/msme-profile-detail?id={msmeId}` OR `/create-business-page?edit={msmeId}` - `/talent-profile-detail?id={talentId}` OR `/create-talent?edit={talentId}` - `/job-detail?id={jobId}` OR `/manage-job?id={jobId}` - `/freelancer-profile?id={freelancerId}` OR `/create-freelancer-page?edit={freelancerId}`

These links let the user immediately act (edit, update, view analytics, etc.).

---

6. Error Handling / Resilience

6.1 Network / Auth Failures

- If any fetch 401s:
- We should NOT crash the entire module.
- We can:
    - log out user / redirect to login, OR
    - show "Please log in again" inline for that specific block.
- If any fetch 500s:
- Show graceful fallback:
    - "We couldn't load your [Business Pages] right now. Please try again later."
- Do **not** render raw JS/stack traces.

6.2 Partial Success

- It's normal for the user to have MSME data but zero Talent data.
- Each block is independently safe to render.

6.3 Image Reliability

- All rows that include logos / profile photos must implement fallback avatars.
- Broken images shouldn't visually break the table row height.

---

7. Security / Permissions Model

7.1 Ownership Assumption

- All lists are **scoped to the logged-in user**.
- Backend filters by auth token → only return resources created/owned by that user.

7.2 Sensitive Fields

- Internal analytics (views, private stats) SHOULD NOT be surfaced here unless explicitly allowed.

- Public-only data (name, title, logo, short description) is safe.

7.3 Premium Surfaces

- Badges like `isVerified`, `Premium`, `Featured` should only appear if:
- The user actually bought the plan,
- OR the MSME/freelancer/talent was featured by admin.
- Never spoof premium status in UI if backend doesn't confirm.

---

8. RAG / Knowledge Ingestion Design for "My Pages"

This section defines how we ingest "My Pages" domain knowledge into retrieval-augmented generation so AI assistants can: - Answer "Why is My Pages empty?" - Coach a user on next steps ("You haven't created a Talent Profile yet → click Create Talent Page") - Summarize a user's presence across MSME / Talent / Jobs / Freelancer - Power in-product copilots, onboarding chat, and help search

8.1 Namespaces / Indices

We propose a dedicated RAG namespace for modules that are personal-dashboard oriented:

- **Namespace:** `my_pages`
- Sub-namespace / type: `ownership_summary`
- Visibility: **private per user** (scoped by userId)
- Refresh trigger: whenever `/my-pages/` successfully loads OR user creates/updates/deletes an MSME/Talent/Freelancer profile or Job post.

Other platform-level namespaces (for consistency): - `faq` (public help content) - `talent_profile_public` (public talent cards) - `msme_public` (public MSME listings) - `jobs_public` (public job posts) - `timeline_public` (public timeline content)

> `my_pages` is special because it's *not* public. It is personalized and should NEVER leak to other users.

8.2 Document Shape for RAG Chunks

For each user, we ingest ONE synthesized document that captures the snapshot of "My Pages" at ingestion time.

Example RAG document body (pseudocode JSON-ish, but stored as plain text in the vector DB):

```
USER_ID: user_456
LAST_SYNCED: 2025-10-29T22:55:00Z

BUSINESS_PAGES:
- msme_123 | "Acme Design Studio" | Industry: Creative Services | Location:
Mumbai | Verified: true
- msme_999 | "NorthStar Analytics" | Industry: Data Consulting | Location:
Remote | Verified: false
```

```
TALENT_PROFILES:
- tal_987 | "Rhea Kapoor" | Category: Motivational Speaker | Pitch:
"Leadership talks for growth-stage teams"

JOBS_POSTED:
- job_456 | "Content Strategist (Remote)" | Status: Open | PostedAt:
2025-10-20T09:30Z | Location: Remote / Bangalore Preferred

FREELANCER_PROFILE:
- free_222 | "Ishan Mehta" | Headline: "Full Stack Developer (React/Node)" |
YOE: 4 | Skills: React, Node.js, REST APIs, AWS

EMPTY_STATES:
- hasBusinessPage: true
- hasTalentProfile: true
- hasJobs: true
- hasFreelancerProfile: true

USER_GAPS:
- none

RECOMMENDED_NEXT_ACTIONS:
- "Update job_456 if role is already filled"
- "Highlight 'Full Stack Developer' profile in Premium search results"
```

If user had NO Talent Profile yet, the chunk would include:

```
EMPTY_STATES:
- hasTalentProfile: false
USER_GAPS:
- "User has not created a Talent Profile yet"
RECOMMENDED_NEXT_ACTIONS:
- "Prompt user to click 'Create Talent Page' to get discovered for events"
```

Why this structure matters: - AI assistant can answer "What can I still create on ConnecWrk?" using `USER_GAPS`. - AI can generate step-by-step onboarding ("Go to My Pages → click 'Create Freelancer Profile'"). - AI can auto-surface crosslisting upsell ("You posted jobs but don't have an MSME page yet; creating one increases trust").

8.3 Chunking Strategy

- We ingest ONE chunk per user for My Pages because the content volume is relatively small. This chunk is ~1–3 KB text, easy to embed as one vector.
- If a user ends up with *dozens* of MSME pages or *hundreds* of jobs, we shard:
- Chunk A: `BUSINESS_PAGES`
- Chunk B: `TALENT_PROFILES`
- Chunk C: `JOBS_POSTED`
- Chunk D: `FREELANCER_PROFILE`
- Chunk E: `USER_GAPS` / `RECOMMENDED_NEXT_ACTIONS`

Sharding rule of thumb: keep each chunk <4KB tokens equivalent.

8.4 Retrieval Keys / Metadata for Indexing

Every RAG-ingested doc from `my_pages` namespace must include metadata fields for safe filtering:

- `namespace`: "my_pages"
- `userId`: string (owner of this data)
- `lastSynced`: ISO timestamp
- `sectionPresence`:
- `hasBusinessPage` (bool)
- `hasTalentProfile` (bool)
- `hasJobs` (bool)
- `hasFreelancerProfile` (bool)
- `gaps`: array of unmet areas (e.g. `["talent_profile_missing", "freelancer_profile_missing"]`)
- `planTier`: user's premium tier if known (Basic / Basic Plus / Premium), for contextual suggestions

Important: assistants MUST filter retrieval by `userId` before using this content in responses. That enforces privacy.

8.5 "Question → Retrieval" Examples

- Q: "What have I already created on ConnecWrk?"
- Filter RAG: namespace=`my_pages`, userId=`currentUserId`

- Return list of MSME, Talent, Jobs, Freelancer from chunk

- Q: "Why does My Pages say I haven't posted any jobs yet?"

- RAG doc contains `hasJobs=false` → assistant can answer "Because you haven't posted a job yet. You can post via ..."

- Q: "How do I create a Talent Profile?"

- Assistant can answer from `USER_GAPS` + known onboarding flows:
    - Navigate to My Pages → Talent Profiles tab → follow Create Talent Page.

8.6 Redaction / PII Policy for RAG Storage

- We DO allow storing company names, job titles, profile headlines because the AI assistant needs them to help the **same** user.
- We MUST NOT store:
- Personal phone numbers
- Personal emails that aren't already public on their profile
- Salary / rate numbers unless explicitly marked public-facing in the platform UI

- Internal notes, admin flags, moderation status

- All timestamps are OK.

- Locations are OK at city/state/country level. Avoid exact street addresses.

8.7 Update Triggers

We should re-ingest / re-generate this `my_pages` RAG snapshot on: - Page load of `/my-pages/` (successful fetch of all 4 lists) - Creation / update / deletion of: - MSME page - Talent profile - Freelancer profile - Job post

This keeps AI coaching current and reduces hallucination risk.

---

9. How "My Pages" Fits Into the Rest of ConnecWrk

9.1 Onboarding Funnel

- If a new user lands here and everything is empty:

- They instantly see ALL the things they *could* become:

    - MSME ("Create Business Page")
    - Talent / Speaker / Artist ("Create Talent Page")
    - Recruiter / Client ("Post Assignment")
    - Freelancer for hire ("Create Freelancer Profile")

- RAG can then say in chat: "You're not visible yet as a Freelancer. Want me to guide you through creating that profile?"

9.2 Growth Loop / Credibility Loop

- MSME page + job post → makes you look like a real hiring entity.
- Talent profile + posted events (future) → makes you bookable.
- Freelancer profile + portfolio → helps you win assignments.

"My Pages" shows the user: "Here is your *public reputation layer* on ConnecWrk." That’s sticky.

9.3 Premium Surfacing

- Premium plans (Basic Plus / Premium) unlock boosts, featured placements, analytics.
- This screen is a perfect funnel to remind: "Upgrade to feature your MSME / Talent / Freelancer more prominently."
- We can store `planTier` in the RAG metadata and have the assistant suggest relevant premium upsells.

---

10. Edge Cases & Future Enhancements

1. **Multiple identities per user**
2. A user may run 3 MSMEs, be a Talent, AND also freelance.
3. UI must gracefully list multiple rows under each section.

4. RAG chunking MUST shard once we exceed safe size (see §8.3).

5. **Draft / unpublished states**

6. If backend returns drafts, we should label them clearly in UI.

7. RAG `USER_GAPS` could say "Talent profile exists but is still incomplete (Draft)." This helps AI guide completion.

8. **Job moderation / closed jobs**

9. Closed or rejected jobs might be hidden or marked visually. We should include a `status` string in RAG so assistant can answer "Yes, you posted a job, but it's closed."

10. **Team / shared ownership**

11. Future roadmap: an MSME may be co-managed by multiple admins.
12. Then endpoint might need `type=self` vs `type=admin` vs `type=collaborator`.

13. RAG metadata should track `role: owner|admin|editor` per entity so assistant can respect permissions.

14. **Analytics surfacing**

15. Future: "Job X got 42 applicants" / "Talent profile viewed 18 times this week".
16. That data is attractive for premium upsells and AI coaching.
17. We MUST gate analytics behind correct permission + planTier before exposing in RAG.

---

11. Developer + AI Assistant Contract Summary

- `/my-pages/` is *the* canonical single-source-of-truth UI for what a user has created on ConnecWrk.
- Frontend pulls from four separate backend verticals (MSME, Talent, Jobs, Freelancer) in parallel.
- All those slices are recombined into one visual dashboard AND into one `my_pages` RAG snapshot for AI.
- The RAG snapshot:
- is private, user-scoped, non-shareable
- stores ownership, gaps, and next-step suggestions
- powers proactive guidance, onboarding nudges, and premium upsell recommendations

If engineering, product, or AI assistants need to answer:

> "What can this user still do to increase visibility / credibility on ConnecWrk?"

...the `my_pages` namespace is exactly where that truth lives.