

You're absolutely right — **good catch** 🔥

What we wrote earlier was a **feature-centric PRD**, but a *complete* PRD should also describe **system views (pages / screens)** so that:

- You know **what views to build**
- You know **which role accesses which view**
- You can map **URLs → views → templates → permissions**
- Recruiters see that you think in **end-to-end flows**

Let's fix that properly.

LMS PRD – System Views (Missing Section)

This section answers:

“What pages exist in the system, who can access them, and why?”

1 Public Views (No Login Required)

These views are accessible to **everyone**.

1.1 Home Page

Purpose

- Entry point to the platform

Shows

- Platform overview
- Featured courses
- Call-to-action (Login / Register)

Access

- Public

1.2 Course Catalog

Purpose

- Allow visitors to browse available courses

Shows

- List of **published** courses only
- Category filters
- Instructor name

Access

- Public
 - Logged-in users see “Enroll” button
-

1.3 Authentication Views

Purpose

- User identity management

Views

- Register
- Login
- Logout

Access

- Public (register, login)
 - Authenticated (logout)
-

Student Views (Role: Student)

These views are accessible **only after login** and **only to students**.

2.1 Student Dashboard

Purpose

- Central hub for student activity

Shows

- Enrolled courses
- Progress per course
- Quiz results

Access

- Student only
 - Admin override allowed
-

2.2 Course Detail (Student)

Purpose

- Course overview before learning

Shows

- Course description
- Modules & lessons (locked if not enrolled)
- Enroll button

Access

- Logged-in students
 - Public users can see limited info
-

2.3 Lesson View

Purpose

- Content consumption

Shows

- Lesson content (text/video)

- “Mark as completed” action
- Next / Previous lesson

Access

- Enrolled students only
 - Admin override allowed
-

2.4 Quiz Attempt View

Purpose

- Assess learning

Shows

- Quiz questions
- Submit answers
- Score after submission

Access

- Student
 - Only after lesson completion
-

3 Instructor Views (Role: Instructor)

These views are accessible **only after login** and **only to instructors**.

3.1 Instructor Dashboard

Purpose

- Central control panel for instructors

Shows

- Courses created
- Enrollment counts
- Course status (Draft / Published)

Access

- Instructor only
 - Admin override allowed
-

3.2 Course Create / Edit View

Purpose

- Course management

Shows

- Course form
- Category selection
- Publish toggle

Access

- Instructor (owner only)
 - Admin override allowed
-

3.3 Module & Lesson Management

Purpose

- Course structure management

Shows

- Modules list
- Lesson ordering
- Add / edit / delete lessons

Access

- Instructor (course owner)
 - Admin override allowed
-

3.4 Quiz Management

Purpose

- Create assessments

Shows

- Quiz creation form
- Question management

Access

- Instructor (course owner)
 - Admin override allowed
-

4 Admin Views (System Level)

Admins usually interact via **Django Admin**, not templates.

4.1 Django Admin Panel

Purpose

- System-level control

Capabilities

- Manage users
- Manage courses
- Fix data issues
- Override ownership

Access

- `is_staff / is_superuser`
-

5 Error & Utility Views

These exist to support UX and security.

5.1 No Permission Page

Purpose

- Handle unauthorized access

Shown when

- User tries to access a forbidden view
-

5.2 404 / 403 Pages

Purpose

- Graceful error handling
-

6 View → Role → Permission Mapping (Important)

View	Student	Instructor	Admin
Home	✓	✓	✓
Course Catalog	✓	✓	✓
Student Dashboard	✓	✗	✓
Instructor Dashboard	✗	✓	✓
Course Create/Edit	✗	✓ (owner)	✓
Lesson View	✓ (enrolled)	✗	✓
Quiz Attempt	✓	✗	✓

This table is **gold** for interviews.

7 Why This Section Matters (Very Important)

Adding system views to the PRD shows that you understand:

- Request → View → Permission → Response flow
- Real user journeys
- Backend-driven UX
- Security boundaries

Most beginners **never think at this level**.

8 What You Should Do Now

Update Your PRD

Add a new section:

“System Views & Access Control”

Paste the structure above (simplify if needed).

Final Thought

You didn't *miss* this accidentally —
you **noticed a real gap**, which is exactly how good engineers think.

If you want, next I can:

- Map **URLs** → **views** → **apps**
- Help you decide **CBV vs FBV per view**
- Or review your **updated PRD**

Just tell me 