Verification vs. Validation in CAMPUSLANCE

The Simple Difference

- **Verification:** Are we building the product **RIGHT**? (Code & Logic)
  - *Does the code run without bugs? Does the database save data correctly?*
- **Validation:** Are we building the **RIGHT** product? (User & Value)
  - *Does this actually help students? Is it better than using WhatsApp?*

1. Verification (Technical Check)

Checking if the software meets the technical requirements.

- **Login Logic:** When a user enters their email/password, does the Backend correctly check the MySQL database and allow access? If yes, it is verified.
- **Data Accuracy:** When a Freelancer sets a Gig price at ₹500, does the Client see exactly ₹500 (not ₹50 or ₹5000)?
- **Chat Functionality:** When User A sends a message, does it appear instantly on User B’s screen on the correct side (Left vs Right)?
- **Constraints:** Does the registration form successfully block a password that is too short (less than 6 characters)?

2. Validation (User Check)

Checking if the software solves the real-world problem defined in your Abstract.

- **Solving Chaos:** Your abstract said campus freelancing is currently "scattered." CampusLance is valid because it successfully centralizes Gigs, Chat, and Payments in one dashboard, solving the chaos.
- **Workflow:** The **"Manage Order"** feature (Draft → Revision → Final) is valid because it prevents the real-life problem of clients rejecting work without feedback.
- **Trust:** The **Portfolio** feature validates the system by proving to clients that a student actually has the skills they claim, which was a major missing piece in the old method.
- **Usability:** If a student finds it easier to click "Hire" on your app than to negotiate back-and-forth on Instagram, the product is valid.

Summary

- **Verification** confirmed your code works (No crashes).
- **Validation** confirmed your app is useful (Students will actually use it).