

Shashank — Task Brief (BHIV HR Platform Build, with Values + MDVP)

Goal

Deliver a working client-facing HR platform where recruiters/clients can:

- Create jobs
- Upload candidate CVs
- View Top-5 shortlists (from Talah's AI agent)
- Submit feedback including Values rubric
- Track MDVP (Minimum Daily Value Push) for continuous progress
- Export reports
- Run entire system in Docker for deployment

Scope of Work

Day 1 — Foundations (Platform & MDVP Kickoff)

- Scaffold API Gateway (FastAPI/Flask/Express — your choice)
- Define data models:
 - Clients
 - Jobs
 - Candidates
 - Feedback
 - ValuesAssessments (Integrity, Honesty, Discipline, Hard Work → 0–5 scale)
 - Events (interviews, offers, etc.)
- Build Endpoints (stubs initially):
 - POST /v1/jobs → create JD
 - POST /v1/candidates:bulk → upload CVs
 - GET /v1/match/{job_id}/top → call Talah's agent for Top-5

- POST /v1/feedback → recruiter feedback + values rubric

MDVP Requirement:

Every day, push at least one working endpoint or feature live (e.g., first day → jobs + candidates creation). No empty commits.

Done when: can create a job, upload CVs, and get dummy Top-5 list back from stub.

Day 2 — Values & Dashboard

- Add Values Rubric UI in client portal.
- Save rubric → attach to candidate profile.
- Extend POST /v1/feedback to accept both free-text + values scores.
- Build Dashboard v1 (Streamlit/React):
 - Candidate funnel (sourced → screened → interviewed → offered → hired)
 - Pie chart of sentiment (from Talah)
 - Bar chart of average candidate values scores per job

MDVP Requirement:

Push a working values feedback form + basic dashboard by EOD. Even if visuals are basic, they must show live values data.

Done when: recruiter can submit values scores & feedback, and dashboard updates automatically.

Day 3 — Scheduling, Offers, Reporting

- Implement Interview Scheduling stub:
 - POST /v1/interviews → record date/time, candidate, job
- Implement Offer Management stub:
 - POST /v1/offers → record sent/accepted/rejected
- Build Report Export:
 - GET /v1/reports/job/{job_id}/export.csv → includes candidate values scores & recruiter sentiment

- Add Auth (API key/token per client)
- Dockerize gateway + portal; add docker-compose.yml with Talah's agent as service

MDVP Requirement:

Deliver 1 working client-facing feature (e.g., offer logs or CSV export) by EOD.

Done when: recruiter can schedule interview, log offers, export job report, and run platform locally with one docker compose up.

Day 4 — Polish & Daily Push Closure

- Harden security:
 - Mask PII (no raw IDs)
 - Restrict API with tokens
 - CORS → client portal only
- Finalize dashboard (trendlines: hires, values growth over time)
- Write README.md with:
 - Setup instructions
 - API contracts (sample payloads)
 - Env variables (.env)
 - Deployment notes
- Write Reflection.md: how Integrity, Honesty, Discipline, Hard Work showed up in the work

MDVP Requirement:

Final daily push should be a secure, documented, working system.

Done when: recruiter can complete end-to-end flow → create job → upload CVs → see shortlist → feedback + values → dashboard → export report.

Deliverables

1. API Gateway with routes working
2. Client Portal (upload JD/CVs, shortlist, feedback + values UI, dashboards)

3. Scheduling + Offer stubs, CSV reports with values
4. Docker setup with services (gateway + agent integration)
5. README + docs for deployment
6. Reflection file (Integrity, Honesty, Discipline, Hard Work, Gratitude)

Acceptance Criteria

- Recruiter can complete end-to-end flow with values included
- Each day produces a Minimum Daily Value Push → one working feature committed & running
- Talah's AI agent is called via internal endpoint, not mocked
- Docker Compose runs entire system locally
- README + Reflection present