

NovaOS Dashboard Progress & Plan Backup

Date: 2025-07-11 17:20 PST

1. Current State

- **LangGraph Agent** is active; Docker services have been largely replaced by LangGraph flows.
- **Redis** status unknown; CLI not found (`redis-cli: command not found`).
- **Socket Server** running locally on port 4000; successfully emits console log `Socket server on 4000` .
- **Next.js Dashboard** scaffolded in `~/Desktop/novaos/dashboard-app` ; running at <http://localhost:3000> .
- **GitHub Repo** present at `git@github.com:soulshiftventures/novaos-dashboard.git` , but push fails due to write-access errors.

2. Major Issues Identified

1. **Redis availability**: Unable to confirm if Redis container/service is running or accessible.
2. **Redis CLI**: `redis-cli` not installed or not on PATH.
3. **Docker**: Uncertainty around what Docker services are currently live (Redis, TrendFetcher, etc.).
4. **GitHub Authentication**: SSH keys and PAT scopes configured, but repository permissions still blocking pushes.

3. Immediate Next Steps (Step 1 Check)

1. List Docker Containers

2. Run from project root (`~/Desktop/novaos`):

```
docker ps
```

3. Confirm `redis` , `trend-fetcher` , or any other NovaOS services are listed.

4. Install or Access Redis CLI

5. If Redis is running locally (not in Docker):

```
brew install redis
# then: redis-cli -h localhost -p 6379
```

6. If Redis is in Docker:

```
docker exec -it <redis-container> redis-cli
```

7. **Smoke-Test pub/sub** via CLI:

8. **Terminal A:**

```
redis-cli SUBSCRIBE novaos:queue
```

9. **Terminal B:**

```
redis-cli PUBLISH novaos:queue '{"text":"hello","category":"test"}'
```

10. **Capture & Share** the CLI outputs for Docker status & Redis pub/sub.

4. Longer-Term Plan (After Step 1)

- **Verify Socket Flow End-to-End:** Redis → Socket Server → Browser client.
- **Consolidate Dashboard Code** into the main NovaOS repo or keep separate as decided.
- **Finalize GitHub Setup:** Ensure SSH remote, PAT, and repo permissions are correct.
- **Implement LangGraph Integration:** Connect channels and flows into the dashboard.
- **Release v0.1:** Observable feed with live updates.
- **Iterate:** Add trend-fetcher, KPI widgets, and business-stream management panels.

Once you share the `docker ps` and Redis CLI outputs, I'll update this backup and guide you through exactly which commands to run, in which terminal, and where you should be at each step.