



# DevHouse SF x Omi

## Builder Instruction Booklet

### Core Mission

Omi is building real-world AI hardware (glasses & pendants) that integrates software, firmware, and AI systems.

Your goal as a builder is to **contribute production-quality improvements** that Omi can realistically ship or build upon after DevHouse.

| ⚠ This is not a toy project.

### 1. What You'll Be Working On

Participants can choose **one** of the following paths:

#### Path A: Bug Fixes (Recommended for Faster Impact)

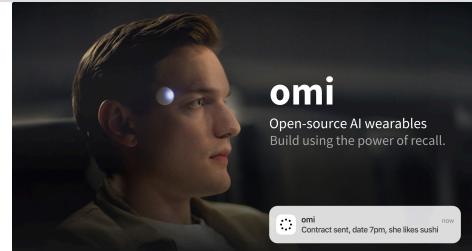
Work on **existing GitHub issues** raised by Omi's core team.

👉 Issues list:

BasedHardware/omi

AI wearables. Put it on, speak, transcribe, automatically -  
BasedHardware/omi

⌚ <https://github.com/BasedHardware/omi/issues>



### Good fit if you want to:

- Fix concrete problems
- Ship PRs quickly
- Demonstrate strong engineering fundamentals

## Path B: Milestone Features (Higher Risk, Higher Reward)

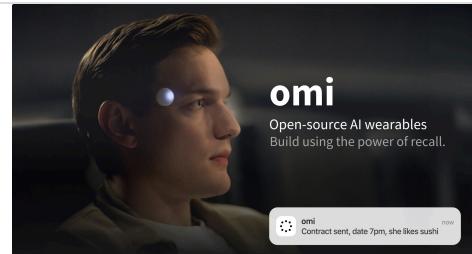
Contribute toward **Milestone 9**, which represents near-term product goals.

👉 Milestone:

BasedHardware/omi

AI wearables. Put it on, speak, transcribe, automatically -  
BasedHardware/omi

⌚ <https://github.com/BasedHardware/omi/milestone/9>



### Good fit if you want to:

- Own a feature end-to-end
- Work closer to product & UX
- Potentially influence roadmap direction

## 2. Hardware Overview (Critical)

Omi has provided **real hardware**:

- AI Glasses
- AI Pendants

These devices:

- Capture audio / contextual data
- Sync with mobile & backend systems
- Rely on stable firmware + software coordination

 **Treat hardware with care**

This is production hardware, not disposable hackathon gear.

### 3. Getting Started (Day 0 Setup)

This is *essential* before making any code changes.

#### Step A — Repo & Branch

```
git clone https://github.com/BasedHardware/omi.git  
cd omi
```

Follow Omi's official branch naming conventions before creating your branch.

#### Step B — Install Dependencies

Follow the docs precisely and confirm:

- hardware SDKs installed
- emulator or simulation mode works
- correct Node / Python / Rust / Android / iOS versions (whatever applies)

If something is unclear:

Document the gap, open a documentation issue — that itself is meaningful contribution.

### 5. Official PR Workflow (Omi Standard)

Omi's docs specify how PRs should be created:

#### 1) Create a Feature Branch

Use a descriptive name:

```
fix/pendant-bluetooth-issue  
feat/milestone9-sync-improvement
```

## 2) Commit Message Standards

Omi's docs align with clear messages:

```
fix: handle disconnection gracefully on glasses
```

## 3) Tests + Validation

Where applicable:

- write tests
- include logs/screenshots
- describe how to reproduce the issue

## 4) Pull Request Review

Your PR must:

- reference the issue it fixes
- explain urgency/impact
- follow the formatting guidelines in the docs

# 6. Working With Hardware

Omi hardware is the **real product**, not a prop.

Your workflow should include:

## A. Hardware Test Plan

Before testing:

- record firmware version
- check battery status

- note environment (WiFi, Bluetooth conditions)

## B. Clear Repro Steps

Example:

1. Put glasses in pairing mode
2. Run app sync
3. Observe log outputs

**Log everything.** If behavior fluctuates, note it.

## 7. Documentation & Quality Matters

Omi's contribution docs emphasize that *documentation is code*.

Therefore:

- update doc pages if behavior changes
- add README references
- improve CLI usage docs if needed

This is a high-value contribution — and reviewers reward clean docs.

## 8. Quality Expectations (Per Omi's Standards)

Your contributions should be:

- ✓ Small and focused
- ✓ Linked to an issue
- ✓ Well-tested
- ✓ Reproducible
- ✓ Review-ready

*If something is ambiguous:*

\*Open an issue asking for clarification — this itself is a contribution.

Omi expects contributors to:

- ✓ Respect coding conventions
- ✓ Write clear, testable code
- ✓ Submit small, focused pull requests
- ✓ Follow the branching & PR guidelines in the docs
- ✓ Assume long-term maintainability

This isn't a hackathon sprint — it's real engineering. *Follow the official doc's expectations.*

## 9. Submission Checklist

Before submitting your PR:

```
## Description
```

```
### What changed
```

<!-- Describe exactly what was modified and where →

```
### Why it matters
```

<!-- Explain the impact on users, hardware, or system behavior →

```
### How it was tested
```

<!--

Include:

- Device **type** (glasses / pendant)
- Firmware version
- App / **OS** version
- Test scenario

→

---

```
## Checklist
```

- [ ] Issue **referenced** (e.g. Fixes #123)

- [ ] Steps to test described clearly
- [ ] Hardware behavior confirmed on real device
- [ ] Logs and/or screenshots provided
- [ ] Code follows repository conventions
- [ ] No hardcoded secrets or debug artifacts

---

## Additional Notes (Optional)

<!-- Any risks, follow-ups, or future improvements -->

## 10. Demo Day & Evaluation

During DevHouse SF Demo Day, be ready to speak to:

1. **Problem you chose**
2. **Why it matters to Omi**
3. **Your solution**
4. **Test results**
5. **Next steps**

If your changes hit the official repo, that's a *strong win signal*.