Montessori Observation Hub (MOH)

Product Requirements Document – v1.0 & v1.5 (June 30 2025)

---

0 Quick view

Item	Detail
Goal	Give every teacher a mobile-first way to capture, browse and report observations; cut report-prep time from hours to minutes.
Launch size	\~50 teachers · 12–14 classrooms · 1–3 admins
Releases	**v1.0** (week 8) – 2-class pilot · **v1.5** (week 12) – full-school roll-out + "quick-win" polish
Must-haves	Google SSO • Voice/text note ≤ 5 min • Tag & date filter • Google-Doc report • Admin bulk roster upload
Future-nice	AI next-steps, parent portal, photos/video attachments

1 Why these choices?

- **Voice limit ≤ 5 min** – matches teachers' real 4-5 min reflections, keeps STT cost predictable and file size sane.
- **Google Docs for reports** – editable before sending; PDF export is one click.
- **Firebase + Google APIs** – fastest path to secure Google SSO, mobile PWA, real-time DB, and push notifications on a free tier.

2 Personas & roles

Persona	Needs	Device
Guide / Assistant Teacher	Quick note capture, easy search, edit within 24 h	Android/iOS phone, Chrome/Safari
Admin / Principal	Cross-class insight, roster import, report generation, analytics	Laptop or phone
Observer (read-only)	View notes without editing (trainee/specialist)	Any

3 Core user flows

```
Teacher  →  Google sign-in
         →  Classroom list  →  Student list  →  Student timeline
         →  Add note  (mic ≤5 min OR type)  →  Pick tags  →  Save

 Admin   →  Google sign-in  →  Admin panel
         →  Bulk upload roster / attendance / assessments (Google Sheet)
         →  Generate report (date range)  →  Google Doc in Drive
         →  Search & filter notes across the school (tag/date/teacher)
```

Link to live sample report template will replace this placeholder once Rahul supplies it.

---

4 Information architecture

4.1 Data model (Firestore)

Collection	Required fields
classrooms	id, name, teacher_ids[]
students	uid (roll no), name, dob, classroom_id, status
observations	id, student_uid, staff_uid, timestamp, text, tags[], starred(bool, default false), edited_at(optional)
attendance	date, student_uid, present(bool)
assessments	id, student_uid, assessment_type(string), result(text), date
tags	id, name, description

Roll-number format `YYYY-CL-###` (e.g. `2025-A2-014`). Algorithm: next unused integer within the same academic year + class; never reused inside that year.

4.2 Access-control rules (Firebase)

```
match /observations/{id} {
  allow create: if request.auth.uid in get(/databases/$(database)/documents/
classrooms/$(classroom)).data.teacher_ids;
  allow update, delete: if request.auth.uid == resource.data.staff_uid &&
                      (request.time - resource.data.timestamp) <
duration.value("24h");
  allow read: if isTeacherForStudent() || isAdmin();
}
```

Teachers read/write only their class; Admins read/write all.

---

5 Functional requirements by version

Area	v1.0 (Week 8)	v1.5 (Week 12)
Authentication	Google SSO	–
Voice/text capture	Mic ≤ 5 min, background STT, inline text edit **+ Quick group (multi-child) selection**	–
Timeline view	Reverse-chrono list; shows every observation for that student from any teacher assigned to the classroom ; Tag & Date filter; Teacher name shown	+ Teacher filter dropdown
Note actions	Edit & Delete within 24 h	–
Tagging	Tag picker; Admin Tag Manager	–
Roster & attendance	Bulk upload via rigid Google-Sheet template	Improved error messages
Reports	Date-range → Google Doc grouped by tag	–
Analytics & dashboards	**Live admin dashboard (obs volume, curriculum heatmap, teacher activity)**	**Tag-usage heatmap (already planned)**
Child insights	**Visual child portfolio (spider chart across curriculum areas)**	**AI-powered next-step suggestions shown beneath each new observation**
CSV export	–	Download filtered notes to CSV
Teacher ↔ Parent	–	**Magic-moment email from starred observation**
Quick-wins	–	Star / unstar + filter; Classroom weekly badge; 10-day inactivity push
–	Star / unstar + filter; Classroom weekly badge; 10-day inactivity push; Tag-usage heatmap	

6 Non-functional requirements

- **Performance** — p95 page load < 1 s on 4 G; upload & transcription start < 10 s for 5 min note.
- **Mobile PWA** — bundle < 150 kB JS, large tap targets, dark mode.
- **Security** — HTTPS, Firestore rule auditing, data encrypted at rest.
- **Retention** — auto-archive student docs 1 yr after leaving; manual delete thereafter.

7 Tech stack

Layer	Tool	Rationale
Front-end	React + Vite PWA	Fast, tiny, offline cache optional
Back-end	Firebase Auth · Firestore · Cloud Functions	Realtime DB, push, Google SSO
STT	Google Speech-to-Text API (en-US)	Lowest friction, cost ≈ ₹80 per 100 min
Docs	Google Docs API	Editable draft in Drive
Charts	`<canvas>` / Chart.js	One small dependency for heatmap

8 Project schedule (96 h cap)

Week	Hours	Deliverables
1	8	Repo, Firebase project, Google SSO working
2	8	Classrooms + Students CRUD, roll-number logic
3	8	Voice capture (5 min cap) + streaming STT
4	8	Student timeline + Tag/Date filters **+ Quick group (multi-child) selection**
5	8	Tag Manager + picker UI
6	8	Bulk roster & attendance upload (happy-path) **+ Live analytics dashboard (obs volume, curriculum heatmap)**
7	8	Report generator (tag-grouped Google Doc) + Edit/Delete window **+ Visual child portfolio widget**
8	8	Pilot in 2 classes, fix critical bugs → **v1.0 ready**
9	8	Star/Unstar + filter · Classroom badge
10	8	10-day inactivity push notification **+ Magic-moment parent email from starred note**
11	8	CSV export (any filter set) **+ AI next-step suggestion layer**
12	8	Tag-usage heatmap · polish uploads · full-school rollout → **v1.5**

Slack time is baked into each week for code review and light QA.

9 Success metrics

Release	Metric	Target
v1.0	Pilot teacher adoption	≥ 80 % teachers log ≥ 2 notes in week 2 pilot

Release	Metric	Target
v1.0	Report speed	Draft Google Doc ≤ 2 min
v1.5	School adoption	$\geq 90\%$ teachers log ≥ 2 notes/week by week 4 post-rollout
v1.5	Data hygiene	Tag heatmap shows $< 10\%$ unused tags

10 Open items (Rahul)

1. Final tag list (CSV or Google Sheet)
2. Sample report template (Google Doc link)
3. Hosting/STT monthly budget ceiling (intern to propose once analytics live)

11 Intern – next 3 concrete tasks

1. **Wireframes** – mock 7 core screens in Figma & review.
2. **Proof-of-concept** – capture 30-s audio, transcribe to Firestore, render in timeline.
3. **Implement** – Classrooms + Students CRUD with sample data.

Document owner: Rahul R. • Last updated: 30 Jun 2025