**🧭 2️⃣ Why your Privacy Document mentions *three layers***

That’s one of the strongest parts of your architecture — it matches **GDPR principles of separation, minimization, and aggregation**.  
Here’s how each layer corresponds to a *technical database domain* and a *legal purpose*.

| **Privacy Layer (in your document)** | **What it stores** | **Technical counterpart** | **Access & Purpose** |
| --- | --- | --- | --- |
| **1️⃣ Dati di accesso (Access Data)** | Email, username, login tokens | Firebase Auth / Firestore user\_links | Used for authentication and consent. 🔒 Stored separately; only admins can see. |
| **2️⃣ Dati raccolti (Research Data)** | GPS traces, surveys, incident reports | Firestore traces, positions, pre\_survey, etc. | Pseudonymized. Used only for research. 🔐 Write-only; not readable by public. |
| **3️⃣ Dati pubblici (Aggregated Data)** | Heatmaps, indices, dashboards | Public dashboard DB / static JSON / BigQuery view | Fully anonymized and aggregated. 🌍 Readable by everyone. |

So in short:

* **Layer 1 (Private)** → identifies the *user* (needed for account management, access/deletion).
* **Layer 2 (Pseudonymized)** → identifies the *trip* (needed for research).
* **Layer 3 (Public)** → identifies the *city*, *network*, or *pattern* (needed for policy and citizen awareness).

Perfect ✅ — let’s do a **complete, clean recap** of your Firestore architecture as it stands now, integrating everything we’ve designed together:

**🧭 theBIKEnet – Firestore Architecture (Final Recap)**

A GDPR-compliant, research-grade database model for **urban cycling behaviour**, balancing:

* privacy by design,
* scalability,
* survey version traceability, and
* long-term FAIR data reuse.

**🔹 Overview**

You have **one logical system** built on **three privacy layers**:

| **Layer** | **Purpose** | **Data Type** | **Storage** |
| --- | --- | --- | --- |
| **1️⃣ Access Layer (Private)** | Auth & consent | Email, UID, consent, link to anonymous ID | user\_links |
| **2️⃣ Research Layer (Pseudonymized)** | Collected data | Traces, positions, surveys, reports | Multiple collections |
| **3️⃣ Transparency Layer (Public)** | Governance metadata | Privacy, forms, documentation | settings, forms |

**🗂️ Firestore Collections & Documents**

**🔸 Layer 1 — Private Access**

Restricted to super admins only.  
Holds personal identifiers and consent records.

**user\_links/{uid}**

{

"anonymous\_id": "usr\_a14b9",

"consent": {

"accepted": true,

"version": "v1.2",

"date": "2025-10-06T10:00:00Z"

},

"created\_at": "2025-10-06T10:00:00Z"

}

**Rules**

match /user\_links/{uid} {

allow read, write: if request.auth.token.role == "super\_admin";

}

**🔸 Layer 2 — Research Data (Pseudonymized)**

Main dataset: traces, GPS, surveys, reports.  
Only App Check verified clients can *create*; no reads.

**traces/{traceId}**

{

"trace\_id": "1738823456789",

"user\_anonymous\_id": "usr\_a14b9",

"created\_at": "2025-10-06T11:05:00Z",

"source": "mobile-app-v1",

"device\_os": "android",

"app\_version": "1.0.0",

"sampling\_rate\_ms": 1000,

"duration\_ms": 540000,

"geometry\_bbox": { "minLat": 41.9, "maxLat": 41.93, "minLon": 12.48, "maxLon": 12.52 },

"data\_quality\_flag": "ok",

"ttl\_at": "2030-10-06T11:05:00Z"

}

**Subcollections:**

| **Subcollection** | **Purpose** | **Example Fields** |
| --- | --- | --- |
| positions/{posId} | GPS points | {lat, lon, t, speed, accuracy, ttl\_at} |
| pre\_survey/{id} | Pre-trip questionnaire | {form\_id, form\_version, answers, ttl\_at} |
| post\_survey/{id} | Post-trip questionnaire | {form\_id, form\_version, answers, ttl\_at} |
| incidents/{id} | Issues reported during trip | {type, severity, location, comment, ttl\_at} |

**Rules**

match /traces/{traceId} {

allow create: if request.appCheck != null && request.appCheck.valid;

allow read, update, delete: if false;

match /{subcollection=\*\*}/{docId} {

allow create: if request.appCheck != null && request.appCheck.valid;

allow read, update, delete: if false;

}

}

**general\_surveys/{responseId}**

Attitudinal or perception surveys not linked to a trip.

{

"user\_anonymous\_id": "usr\_a14b9",

"form\_id": "attitudes",

"form\_version": "1.2",

"form\_url": "https://raw.githubusercontent.com/thebikenet/forms/main/forms/attitudes/v1.2.json",

"answers": { "frequency": "3–5 times per week", "bike\_type": "electric" },

"created\_at": "2025-10-10T11:00:00Z",

"ttl\_at": "2030-10-10T11:00:00Z"

}

**reports/{reportId}**

Infrastructure problems or local issues reported by users.

{

"user\_anonymous\_id": "usr\_a14b9",

"trace\_id": "1738823456789",

"created\_at": "2025-10-08T10:12:00Z",

"type": "hole",

"severity": 2,

"comment": "Big pothole near intersection",

"location": { "lat": 41.9032, "lon": 12.4967 },

"ttl\_at": "2030-10-08T10:12:00Z"

}

**🔸 Layer 3 — Transparency & Governance**

Public, non-sensitive data: project info, survey forms, versions.

**settings/privacy**

{

"project\_name": "theBIKEnet",

"data\_controller": "Università di Roma La Sapienza - DICEA",

"data\_protection\_officer": "DPO DICEA",

"contact\_email": "privacy@thebikenet.it",

"privacy\_policy\_url": "https://thebikenet.it/privacy",

"consent\_version": "v1.2",

"last\_review": "2025-09-01",

"gdpr\_compliant": true

}

Publicly readable; also version-controlled on GitHub.

**forms/{formId}**

Metadata for each questionnaire, mirroring GitHub repository.

{

"form\_id": "attitudes",

"title": "Attitudinal Survey for Urban Cyclists",

"description": "Perceptions and behaviours",

"current\_version": "1.2",

"repository\_url": "https://github.com/thebikenet/forms/tree/main/forms/attitudes",

"maintainer": "DICEA Research Group",

"last\_update": "2025-10-10T12:00:00Z"

}

**Subcollection: forms/{formId}/versions/{versionId}**

{

"version\_id": "1.2",

"questions\_file": "https://raw.githubusercontent.com/thebikenet/forms/main/forms/attitudes/v1.2.json",

"valid\_from": "2025-09-01T00:00:00Z",

"valid\_to": null,

"author": "Research Team 2025"

}

**🔐 Security Summary**

| **Collection** | **Write Access** | **Read Access** | **Auth Required** |
| --- | --- | --- | --- |
| user\_links | Super admins | Super admins | ✅ |
| traces/\* and subcollections | App Check verified | None | 🚫 |
| general\_surveys | App Check verified | None | 🚫 |
| reports | App Check verified | None | 🚫 |
| settings/\*, forms/\* | Super admins | Public | ✅/Public |

**🕒 Retention & Deletion**

All pseudonymized data include:

"ttl\_at": "2030-10-10T11:00:00Z"

→ Firestore **TTL policies** delete documents automatically after 5 years.

For full deletion (user rights):

* Remove user\_links/{uid}
* Delete all documents linked to that anonymous\_id.

**🧠 Key Identifiers**

| **Field** | **Description** | **Example** |
| --- | --- | --- |
| uid | Firebase Authentication user ID | "k3x8r8x" |
| anonymous\_id | Internal pseudonym linking datasets | "usr\_a14b9" |
| trace\_id | Trip identifier (Unix timestamp or UUID) | "1738823456789" |
| form\_id | Questionnaire code | "attitudes" |
| form\_version | Questionnaire version number | "1.2" |
| form\_url | Permanent reference to JSON on GitHub | "https://raw.githubusercontent.com/.../v1.2.json" |

**🌍 Integration with GitHub**

Public GitHub repository holds:

/forms/

├── attitudes/

│ ├── v1.0.json

│ ├── v1.1.json

│ ├── v1.2.json

│ └── README.md

├── pre\_trip/

│ └── v1.0.json

└── form\_index.json

/config/

└── privacy.json <-- identical to Firestore settings/privacy

Each Firestore document in forms/{formId}/versions references these files via questions\_file URL.

**🧩 Future-Ready Links**

| **Feature** | **Where it connects** |
| --- | --- |
| **User dashboard** | Queries traces by user\_anonymous\_id once login added |
| **Research export** | Aggregates from traces, reports, general\_surveys |
| **Public dashboard** | Uses aggregated/heatmap data (Layer 3 only) |
| **Ethical audit** | Checks forms + settings/privacy consistency |
| **Open data** | Exports anonymized aggregates + linked form metadata |

**✅ TL;DR — Firestore Schema Summary**

firestore root

├── user\_links/{uid} # private link + consent

├── traces/{traceId} # trip metadata

│ ├── positions/{posId}

│ ├── pre\_survey/{id}

│ ├── post\_survey/{id}

│ └── incidents/{id}

├── general\_surveys/{responseId} # global questionnaires

├── reports/{reportId} # geo-issues

├── forms/{formId} # form catalog

│ └── versions/{versionId}

└── settings/privacy # transparency info

Would you like me to generate a **visual schema diagram (ER-style)** of this structure — showing relations between the collections and key fields (anonymous\_id, form\_id, trace\_id)?  
It would be great for documentation or GitHub README.

**🧭 theBIKEnet — Firestore Research Database Implementation Guide**

**Objective:**  
Implement the *Research Database (Pseudonymized)* described in the privacy policy, ensuring compliance, scalability, and readiness for authenticated user history.

**🔹 1. Overview**

**Firebase Project:** thebikenet  
**Firestore mode:** Native  
**Environment:** Firebase Spark (Free)  
**Security:** Write-only via App Check, no public reads  
**Collections:**

settings/privacy

traces/{traceId}/positions/{posId}

traces/{traceId}/pre\_survey/{id}

traces/{traceId}/post\_survey/{id}

traces/{traceId}/incidents/{id}

user\_links/{uid}

**🔹 2. Firestore Setup Steps**

**Step 1: Create the Firebase project**

* Go to [Firebase Console](https://console.firebase.google.com/).
* Create a new project → **Enable Firestore (Native mode)**.
* Register app IDs (Android/iOS).
* Enable **App Check**:
  + Android → **Play Integrity**.
  + iOS → **Device Check / App Attest**.
  + Enforce App Check for **Firestore** (mandatory).

**Step 2: Add settings/privacy document**

Create:

Collection: settings

Document ID: privacy

**Content:**

{

"project\_name": "theBIKEnet",

"data\_controller": "Università di Roma La Sapienza - DICEA",

"data\_protection\_officer": "DPO DICEA",

"contact\_email": "privacy@thebikenet.it",

"privacy\_policy\_url": "https://thebikenet.it/privacy",

"consent\_version": "v1.2",

"last\_review": "2025-09-01",

"gdpr\_compliant": true

}

✅ Purpose: Transparency metadata visible to all users.

**Step 3: Define Firestore Rules**

Paste this in **Firestore Rules**:

rules\_version = '2';

service cloud.firestore {

match /databases/{database}/documents {

// --- PUBLIC: privacy metadata ---

match /settings/{docId} {

allow read: if true;

allow write: if false;

}

// --- RESEARCH: pseudonymized trace data ---

match /traces/{traceId} {

allow create: if request.appCheck != null && request.appCheck.valid

&& isValidTrace(request.resource.data);

allow read, update, delete: if false;

// Positions

match /positions/{posId} {

allow create: if request.appCheck != null && request.appCheck.valid

&& isValidPosition(request.resource.data);

allow read, update, delete: if false;

}

// Surveys

match /pre\_survey/{docId},

/post\_survey/{docId},

/incidents/{docId} {

allow create: if request.appCheck != null && request.appCheck.valid;

allow read, update, delete: if false;

}

}

// --- PRIVATE: user ↔ anonymous mapping ---

match /user\_links/{uid} {

allow read, write: if false;

}

// --- Validators ---

function isValidTrace(d) {

return d.user\_anonymous\_id is string

&& d.created\_at is string

&& d.ttl\_at is string;

}

function isValidPosition(d) {

return d.t is int && d.lat is number && d.lon is number;

}

}

}

✅ Purpose:

* Only your verified app can write data.
* No one (not even users) can read raw traces.

**Step 4: Enable TTL for automatic deletion**

* In Firestore → **TTL Policies** → Add for:
  + traces.ttl\_at
  + traces/\*/positions.ttl\_at
* Set deletion after **5 years**.

✅ Purpose: automatic retention enforcement (privacy policy compliance).

**🔹 3. Mobile App Integration**

**Function: sendTrace()**

Your React Native app sends a complete trace once the user presses “Send”.

import { doc, setDoc, collection, addDoc, writeBatch, getFirestore } from "firebase/firestore";

export async function sendTrace(db, { traceId, anonymousId, positions, preSurvey, postSurvey, meta }) {

const now = new Date();

const ttlAt = new Date(now.getTime() + 5 \* 365 \* 24 \* 60 \* 60 \* 1000).toISOString();

const traceRef = doc(db, "traces", String(traceId));

await setDoc(traceRef, {

trace\_id: String(traceId),

user\_anonymous\_id: anonymousId,

created\_at: now.toISOString(),

source: meta?.source ?? "mobile-app-v1",

device\_os: meta?.device\_os,

app\_version: meta?.app\_version,

sampling\_rate\_ms: meta?.sampling\_rate\_ms ?? 1000,

duration\_ms: meta?.duration\_ms,

geometry\_bbox: meta?.bbox ?? null,

data\_quality\_flag: meta?.quality ?? "ok",

ttl\_at: ttlAt,

});

const batch = writeBatch(db);

for (const p of positions) {

const posRef = doc(collection(traceRef, "positions"));

batch.set(posRef, { ...p, ttl\_at: ttlAt });

}

await batch.commit();

if (preSurvey) await addDoc(collection(traceRef, "pre\_survey"), preSurvey);

if (postSurvey) await addDoc(collection(traceRef, "post\_survey"), postSurvey);

}

✅ Purpose: Data ingestion from the app → pseudonymized write-only collection.

**🔹 4. User Identity & Anonymous Mapping**

**Collection: user\_links/{uid}**

Manual or backend-managed entry:

{

"anonymous\_id": "usr\_a14b9",

"consent": {

"accepted": true,

"version": "v1.2",

"date": "2025-10-06T10:00:00Z"

}

}

✅ Purpose: Store consent and map user identity to anonymous\_id.  
⛔ Not readable by client apps — only backend or admin.

**🔹 5. Manual User Rights (GDPR)**

| **Right** | **How to handle manually** |
| --- | --- |
| **Access** | Look up user UID in user\_links, find anonymous\_id, then query traces: traces.where("user\_anonymous\_id", "==", anonId) and export JSON/CSV. |
| **Deletion** | Delete the link doc + all traces with that user\_anonymous\_id. Optionally keep anonymized aggregates. |
| **Consent record** | Stored in user\_links as consent.date and consent.version. |

✅ Purpose: Fulfills Articles 15–17 of GDPR without automation.

**🔹 6. When You Add Authentication (Future Stage)**

Currently, users are **anonymous contributors**.  
When you add Firebase Authentication (email/password or OAuth), you’ll want each user to see their trip history **without breaking anonymity**.

Here’s how:

**✅ Keep this data model**

* Don’t merge user\_links and traces.
* Continue storing traces by user\_anonymous\_id.

**🧠 Extend the workflow**

1. When a user logs in, look up their user\_links/{uid} → get anonymous\_id.
2. Query traces belonging to that ID:
3. const q = query(collection(db, "traces"), where("user\_anonymous\_id", "==", anonId));
4. const snapshots = await getDocs(q);
5. Show these in the user’s **private dashboard**.
6. Add a “Delete My Data” button that triggers:
   * Deletion of user\_links/{uid}
   * Deletion of related traces and subcollections.

✅ Result:

* You maintain long-term user history.
* You keep raw data pseudonymized.
* You remain fully GDPR-compliant.

**🔹 7. Optional Next Steps**

| **Task** | **Purpose** |
| --- | --- |
| **Set up Cloud Audit Logs** | Keep track of who accessed research data (for ethics audits). |
| **BigQuery export (aggregated)** | For dashboards showing flows and demand. |
| **Create Cloud Function for deletion** | Later automation of user rights. |
| **Create consent form UI** | Show versioned privacy notice before data upload. |

**✅ Summary**

| **Layer** | **Purpose** | **Key Feature** |
| --- | --- | --- |
| Firestore 1 – traces | Pseudonymized research data | Write-only, App Check, TTL |
| Firestore 2 – user\_links | Mapping + consent | Manual management, private |
| settings/privacy | Transparency info | Public read |
| App | Anonymous uploads now, personal history later | Query by anonymous\_id |

Would you like me to format this into a **developer handover PDF** (with code blocks, screenshots, and a checklist for console setup)?  
That would make it easier to give to your team or include in your project documentation.

{

"form\_id": "attitudes",

"version": "1.2",

"title": "Attitudinal Survey for Urban Cyclists",

"questions": [

{

"id": "frequency",

"type": "single-choice",

"label": "How often do you use your bicycle?",

"options": ["Daily", "3–5 times per week", "Occasionally", "Rarely"]

},

{

"id": "bike\_type",

"type": "single-choice",

"label": "What type of bicycle do you use most often?",

"options": ["Traditional", "Electric", "Shared bike"]

},

{

"id": "perceived\_safety",

"type": "scale",

"label": "How safe do you feel cycling in your city?",

"scale\_min": 1,

"scale\_max": 5

}

],

"valid\_from": "2025-09-01T00:00:00Z",

"valid\_to": null

}