

aayu-mitra: Firebase Data Structure (v2)

This document defines the complete Firestore database structure for the simplified "v2" of the aayu-mitra project. This version includes the Raspberry Pi (LLM Assistant) and the Flutter App, but **does not** include the Autoencoder ML pipeline.

This is the API Contract your team must follow.

Root Path

All collections are rooted at: /artifacts/{appId}/public/data/

1. user_profile (Collection)

Stores the core, non-changing data for a single user/device hub. This is the LLM's "long-term memory."

- **Document ID:** {pi_id} (e.g., "pi-hub-001")
- **Data Model:**

```
{
  "name": "string", // e.g., "Elderly User Name"
  "age": "number",
  "notes": "string", // General notes for the LLM
  "sleep_schedule": {
    "planned_bedtime": "string (HH:MM)", // From "Set Your Sleep Routine"
    "planned_wakeup": "string (HH:MM)"
  }
}
```

2. medication (Collection)

Stores a list of all recurring medications for a user. Matches your "Medication Routine" and "Add Medication" screens.

- **Document ID:** Auto-generated ({medication_id})
- **Data Model:**

```
{
  "pi_id": "string", // Foreign key to user_profile (e.g., "pi-hub-001")
  "title": "string", // e.g., "Paracetamol"
  "category": "string", // e.g., "Pill", "Syrup"
  "amount": "string", // e.g., "2 pills", "2 tablespoons"
  "time_slots": "array<string>", // ["Morning", "Afternoon", "Night"]
  "repeat_days": "array<string>" // ["Monday", "Wednesday", "Daily"]
}
```

3. routines (Collection)

Stores recurring non-medication routines. Matches your "Routine Page" and "Add New Routine" screens.

- **Document ID:** Auto-generated ({routine_id})
- **Data Model:**

```
{
  "pi_id": "string", // Foreign key
  "title": "string", // e.g., "Morning Walk"
  "time": "string (HH:MM)", // e.g., "6:00 AM"
  "repeat_days": "array<string>", // ["Monday", "Wednesday",
  "Friday"]
  "notes": "string" // e.g., "Walk for 30 minutes in the park."
}
```

4. reminders (Collection)

Stores one-off, non-recurring reminders. Matches your "Add Reminder" screen.

- **Document ID:** Auto-generated ({reminder_id})
- **Data Model:**

```
{
  "pi_id": "string", // Foreign key
  "label": "string", // e.g., "Doctor's Appointment"
  "details": "string", // e.g., "Annual checkup"
  "date": "string (YYYY-MM-DD)",
  "time": "string (HH:MM)",
  "priority": "string ('Low', 'Normal', 'High')",
  "completed": "boolean" // Default: false
}
```

5. health_records (Collection)

A time-series log of the user's vitals. Matches your "Track Health" and "Enter Health Details" screens.

- **Document ID:** Auto-generated ({record_id})
- **Data Model:**

```
{
  "pi_id": "string", // Foreign key
  "date": "timestamp", // Use ServerTimestamp for sorting
  "weight_kg": "number",
  "blood_pressure": "string", // e.g., "120/80"
  "heart_rate_bpm": "number", // e.g., 72
  "temperature_f": "number" // e.g., 98.6
}
```

6. sleep_logs (Collection)

A time-series log of the user's sleep. Matches your "Sleep Routine" screen's "Weekly Sleep Cycle" chart.

- **Document ID:** Auto-generated ({log_id})
- **Data Model:**

```
{
  "pi_id": "string", // Foreign key
  "date": "string (YYYY-MM-DD)", // The date the sleep *ended*
(wake-up day)
  "duration_hours": "number" // e.g., 6.9, 8, 5
}
```

7. alerts (Collection)

The main feed for all system-generated alerts. Matches your "Notifications" screen.

- **Document ID:** Auto-generated ({alert_id})
- **Data Model:**

```
{
  "pi_id": "string", // Foreign key
  "level": "string ('high', 'info')", // 'high' for "I feel dizzy"
  "source": "string ('Pipeline_A_LLM')", // Source is the LLM
  "message": "string", // e.g., "User reported feeling dizzy."
  "timestamp": "timestamp", // Use ServerTimestamp
  "acknowledged": "boolean" // Default: false
}
```

8. messages (Collection)

The complete chat log. Matches your "Conversation History" and "Leave a Message" screens.

- **Document ID:** Auto-generated ({message_id})
- **Data Model:**

```
{
  "pi_id": "string", // Foreign key
  "sender": "string ('Mitara', 'Elderly', 'Caregiver')",
  "text": "string", // The content of the message
  "timestamp": "timestamp" // Use ServerTimestamp
}
```