Backend Developer - Firebase Test

Overview

Medverse builds PC training simulations for medical devices. Our backend runs on Firebase (Cloud Functions, Firestore) with TypeScript.
This test evaluates your ability to set up Firebase infrastructure and write clean TypeScript Cloud Functions.

Requirements

Project Setup

- [ ] Create a new Firebase project
- [ ] Initialize Cloud Functions with TypeScript
- [ ] Configure Firestore database
- [ ] Set up proper project structure

Cloud Functions

1. Create Session

- [ ] HTTP endpoint to create a new session
- [ ] Generates unique session ID
- [ ] Accepts `region` parameter (e.g., "eu-central", "us-east")
- [ ] Stores in Firestore with:
  - `sessionId`
  - `region`
  - `status` (default: "pending")
  - `createdAt` timestamp

- `updatedAt` timestamp

- [ ] Returns the created session object

2. Get Session

- [ ] HTTP endpoint to retrieve a session by ID
- [ ] Returns session data or appropriate error if not found

3. Update Session Status

- [ ] HTTP endpoint to update a session's status
- [ ] Valid statuses: "pending", "active", "completed", "failed"
- [ ] Validates status value
- [ ] Updates `updatedAt` timestamp

Code Quality

- [ ] Clean TypeScript (proper types, no `any`)
- [ ] Consistent error handling
- [ ] Input validation
- [ ] Sensible code organization

Documentation

- [ ] README with:
  - Setup instructions
  - How to deploy
  - API endpoint documentation
  - Any decisions or trade-offs you made

Bonus (Optional)

If you have additional time and want to demonstrate more depth:

☐ **Authentication** - Protect endpoints with Firebase Auth, API key validation, or JWT

☐ **Unit Tests** - Add tests for your functions

☐ **List Sessions** - Endpoint to list sessions with optional filtering by status or region

---

Deliverables

1. **Firebase project** - Share access via Firebase console (email: s.dieckmann@medverse.de)
2. **Code repository** - GitHub repo with your code
3. **README** - Setup and documentation
4. **Brief notes** - Any questions, assumptions, or trade-offs you made

---

Evaluation Criteria

Area	What We're Looking For
TypeScript	Clean types, interfaces, no shortcuts
Firebase/Firestore	Sensible data modeling, proper use of Firebase patterns
Error Handling	Graceful failures, appropriate HTTP status codes
Code Organization	Logical structure, separation of concerns
Documentation	Clear setup instructions, API docs
Bonus Items	Security thinking, testing approach (if attempted)

Getting Started

1. Create a new Firebase project at console.firebase.google.com
2. Enable Firestore and Cloud Functions
3. Initialize with `firebase init` (select Functions + Firestore, TypeScript)
4. Build and test locally with the Firebase emulator

5. Deploy when ready

Questions?

If anything is unclear, reach out to email: s.dieckmann@medverse.de. We're happy to clarify requirements - asking good questions is part of the evaluation.