# **🛠 Centsible – Maintenance User Guide**

## **📁 Project Overview**

**Centsible** is a Next.js-based web application that allows users to track their subscriptions, generate reports, and visualize data using charts. It uses Firebase Authentication and Firestore for user management and data storage.

## **🧰 Requirements**

### **Software**

* **Node.js** (v18 or higher)
* **npm** or **yarn**
* **Firebase** account
* **Git**
* Optional: **VS Code**, **Tailwind CSS IntelliSense** extension

## **🔧 Setup Instructions**

### **1. Clone the Repository**

git clone https://github.com/mikebutts/centsible.git

cd centsible

### **2. Install Dependencies**

npm install

*or*

yarn install

### **3. Firebase Configuration**

#### **A. Firestore & Authentication**

1. Go to [Firebase Console](https://console.firebase.google.com/).
2. Create a project (or use an existing one).
3. Enable **Authentication** (Email/Password sign-in).
4. Enable **Firestore Database**.

#### **B. Set Up Firebase Credentials**

In your project root, create a .env.local file and add the following:

NEXT\_PUBLIC\_FIREBASE\_API\_KEY=your-api-key

NEXT\_PUBLIC\_FIREBASE\_AUTH\_DOMAIN=your-project-id.firebaseapp.com

NEXT\_PUBLIC\_FIREBASE\_PROJECT\_ID=your-project-id

NEXT\_PUBLIC\_FIREBASE\_STORAGE\_BUCKET=your-project-id.appspot.com

NEXT\_PUBLIC\_FIREBASE\_MESSAGING\_SENDER\_ID=sender-id

NEXT\_PUBLIC\_FIREBASE\_APP\_ID=app-id

⚠️ These are **client-side** environment variables.

If you are using firebase-admin, add the GOOGLE\_APPLICATION\_CREDENTIALS and base64-encoded key if needed for server functions.

### **4. Configure Firestore Rules**

Update Firestore Security Rules to:

rules\_version = '2';

service cloud.firestore {

match /databases/{database}/documents {

match /subscriptions/{docId} {

allow read, write: if request.auth != null && request.auth.uid == resource.data.userId;

}

match /reports/{docId} {

allow read, write: if request.auth != null && request.auth.uid == resource.data.userId;

}

}

}

## **🚀 Running the App Locally**

npm run dev

Visit: [http://localhost:3000](http://localhost:3000/)

## **🧪 Testing (Manual QA)**

* **Login/Register**: Test authentication and dashboard loading
* **Add/Edit/Delete Subscriptions**: Validate Firebase updates
* **Report Page**: Test filters and charts
* **PDF Export**: (if enabled) Verify report export functionality

## **🛠 Maintenance Tips**

* **Subscriptions Logic**: Found in context/AuthContext.tsx
* **Reports Filtering**: Located in /reports/page.tsx
* **Premium vs Standard**: Controlled by a premium: boolean field in subscription entries
* **Styling**: Tailwind CSS classes used throughout

## **🧼 Troubleshooting**

| **Problem** | **Solution** |
| --- | --- |
| "Missing or insufficient permissions" | Check Firestore rules |
| Blank charts | Verify correct cost and category fields |
| Firebase connection issues | Check .env.local and Firebase config |
| Error: self is not defined | Only use client-side code with 'use client' directive |

## **📦 Deployment**

If deploying to **Vercel**:

1. Push your repo to GitHub.
2. Connect your repo in [Vercel](https://vercel.com/dashboard).
3. Set environment variables in Vercel → Project Settings → Environment Variables.