

EcoShower - Architecture Document

מסמך ארכיטקטורה מפורט

1. סקירה כללית (Overview)

EcoShower מערכת מנטרת את AWS בענן Serverless חכמה לחיסכון במים, המבוססת על ארכיטקטורת IoT היא מערכת המערכת טמפרטורת המים בצנרת, מבצעת סירקולציה אוטומטית של מים קרים, ומתריעה למשתמש כאשר המים מוכנים.

2. רכיבי המערכת

2.1 רכיב הקצה (Edge Device) - ESP32 Controller

תפקיד: איסוף נתוני טמפרטורה ושליטה בברז האלקטרוני

רכיבי חומרה:

- ESP32 DevKit - מובנה WiFi מיקרו-בקור עם
- DS18B20 - חיישן טמפרטורה עמיד למים
- Servo Motor / Solenoid Valve - שליטה בזרימת המים
- Power Supply 5V

תקשורת:

- פרוטוקול: MQTT over TLS (port 8883)
- QoS Level: 1 (At least once delivery)
- Topics:
 - `ecoshower/{device_id}/telemetry` - שליחת נתונים
 - `ecoshower/{device_id}/commands` - קבלת פקודות

2.2 AWS IoT Core

IoT-תפקיד: ניהול התקשורת עם מכשירי ה

הגדרות:

- Thing Type: `EcoShowerDevice`
- Thing Policy: מוגבלות `publish/subscribe` הרשאות
- Certificate: לכל מכשיר X.509

IoT Rules:

```
SELECT * FROM 'ecoshower/+/telemetry'
WHERE temperature > 0
```

Actions:

1. Invoke Lambda function **EcoShower-ProcessTelemetry**
 2. Store to DynamoDB table **EcoShower-RawData**
-

2.3 AWS Lambda Functions

2.3.1 EcoShower-ProcessTelemetry

- **Trigger:** IoT Rule
- **Runtime:** Python 3.11
- **Memory:** 256 MB
- **Timeout:** 30 seconds
- **תפקיד:** עיבוד נתוני טלמטריה, בדיקת סף טמפרטורה, שליחת התראות

2.3.2 EcoShower-API-Users

- **Trigger:** API Gateway
- **Runtime:** Python 3.11
- **Memory:** 256 MB
- **תפקיד:** (CRUD) ניהול משתמשים

2.3.3 EcoShower-API-Devices

- **Trigger:** API Gateway
- **Runtime:** Python 3.11
- **תפקיד:** ניהול מכשירים ופרופילים

2.3.4 EcoShower-API-Dashboard

- **Trigger:** API Gateway
- **Runtime:** Python 3.11
- **תפקיד:** אחזור נתוני חיסכון ודשבורד

2.3.5 EcoShower-SendCommand

- **Trigger:** API Gateway
 - **Runtime:** Python 3.11
 - **תפקיד:** IoT Core שליחת פקודות למכשיר דרך
-

2.4 Amazon DynamoDB

טבלאות:

1. EcoShower-Users

Attribute	Type	Description
-----------	------	-------------

Attribute	Type	Description
user_id (PK)	String	UUID
email	String	כתובת אימייל (GSI)
password_hash	String	סיסמה מוצפנת
role	String	admin / user
name	String	שם מלא
created_at	String	ISO timestamp

2. EcoShower-Devices

Attribute	Type	Description
device_id (PK)	String	UUID
user_id (GSI)	String	בעל המכשיר
name	String	שם המכשיר
target_temp	Number	טמפרטורת יעד
status	String	online/offline
last_seen	String	ISO timestamp

3. EcoShower-Telemetry

Attribute	Type	Description
device_id (PK)	String	מזהה מכשיר
timestamp (SK)	String	ISO timestamp
temperature	Number	טמפרטורה בצלזיוס
water_saved	Number	ליטרים שנחסכו
status	String	heating/ready/idle

4. EcoShower-Sessions

Attribute	Type	Description
session_id (PK)	String	UUID
device_id (GSI)	String	מזהה מכשיר
start_time	String	תחילת מקלחת
end_time	String	סיום מקלחת
water_saved	Number	ליטרים שנחסכו

Attribute	Type	Description
money_saved	Number	(₪) כסף שנחסך

2.5 Amazon API Gateway

Type: REST API

Endpoints:

```

/auth
  POST /login      - התחברות
  POST /register   - הרשמה
  POST /logout     - התנתקות

/users
  GET /            - רשימת משתמשים (admin)
  GET /{id}        - פרטי משתמש
  PUT /{id}        - עדכון משתמש
  DELETE /{id}     - מחיקת משתמש (admin)

/devices
  GET /            - רשימת מכשירים
  POST /           - הוספת מכשיר
  GET /{id}        - פרטי מכשיר
  PUT /{id}        - עדכון מכשיר
  DELETE /{id}     - מחיקת מכשיר
  POST /{id}/command - שליחת פקודה

/dashboard
  GET /summary     - סיכום חיסכון
  GET /history     - היסטוריית מקלחות
  GET /realtime/{id} - נתונים בזמן אמת

/admin
  GET /stats       - סטטיסטיקות מערכת
  GET /users       - ניהול משתמשים
  GET /devices     - כל המכשירים

```

Authorization:

- Cognito User Pools עם JWT tokens
- Custom Lambda Authorizer לבדיקת הרשאות

2.6 Amazon Cognito

User Pool: EcoShower-Users

- Self-registration: Enabled

- Email verification: Required
- Password policy: 8+ chars, uppercase, number, symbol

Groups:

- **admins** - מנהלי מערכת
- **users** - משתמשים רגילים

App Client: EcoShower-WebApp

- OAuth flows: Authorization code grant
 - Token validity: Access 1hr, Refresh 30 days
-

2.7 Amazon SNS

Topic: Dynamic Per-User Topics (**EcoShower-User-`{uuid}`**)

Privacy Design: Unlike traditional systems using a single global topic, EcoShower creates a **Private SNS Topic** for each user upon registration.

- **Isolation:** Users only receive alerts for their own devices.
- **Control:** The backend checks user preferences (**water_ready_alert**) before publishing to this private topic.

Subscriptions:

- Email notifications (Confirmed via link)
- SMS (optional)

Message Types:

- **WATER_READY** - המים חמים
 - **DEVICE_OFFLINE** - מכשיר לא מגיב
 - **MAINTENANCE_ALERT** - התראת תחזוקה
-

2.8 Amazon S3

Bucket: ecoshower-frontend-`{account-id}`

Purpose: Frontend אחסון קבצי (Static Website Hosting)

Configuration:

- Static website hosting: Enabled
 - Public access: Via CloudFront only
 - CORS: Configured for API calls
-

2.9 Amazon CloudFront

Distribution: Frontend CDN

Origins:

1. S3 bucket (frontend)
2. API Gateway (API calls)

Behaviors:

- `/api/*` → API Gateway
- `/*` → S3 bucket

3. זרימת נתונים (Data Flow)

3.1 תרחיש: משתמש מפעיל מקלחת

1. באפליקציה "Start" משתמש לוחץ
↓
2. Frontend → API Gateway → Lambda (SendCommand)
↓
3. Lambda → IoT Core → MQTT → ESP32
↓
4. ESP32 מתחיל לנטר טמפרטורה
↓
5. ESP32 → MQTT → IoT Core → IoT Rule → Lambda (ProcessTelemetry)
↓
6. Lambda ובודק טמפרטורה DynamoDB-שומר ב
↓
7. אם טמפרטורה \geq יעד:
 למשתמש `Lambda → SNS → Push Notification`
 (פתיחת ברז) `Lambda → IoT Core → ESP32`

3.2 תרחיש: משתמש צופה בדשבורד

1. Frontend Dashboard טוען דף
↓
2. API Call: GET /dashboard/summary
↓
3. API Gateway → Lambda → DynamoDB Query
↓
4. Lambda מחשב סיכומים ומחזיר JSON
↓
5. Frontend מציג גרפים וסטטיסטיקות

4. אבטחה (Security)

4.1 שכבות אבטחה

1. Network Level:

- HTTPS only (TLS 1.2+)
- CloudFront WAF

2. Authentication:

- Cognito JWT tokens
- Token refresh mechanism

3. Authorization:

- Role-based access control (RBAC)
- Lambda Authorizer

4. Data:

- Encryption at rest (DynamoDB, S3)
- Encryption in transit (TLS)

5. IoT:

- X.509 certificates per device
- Device policies (least privilege)

4.2 IAM Roles

- `EcoShower-LambdaExecutionRole` - הרשאות ל-Lambda functions
- `EcoShower-IoTDeviceRole` - הרשאות למכשירי IoT
- `EcoShower-APIGatewayRole` - הרשאות ל-API Gateway

5. Scalability & Performance

5.1 Auto-scaling

- Lambda: Automatic (up to 1000 concurrent)
- DynamoDB: On-demand capacity mode
- API Gateway: Automatic scaling

5.2 Performance Optimizations

- DynamoDB DAX for caching (optional)
- CloudFront caching for static assets
- Lambda provisioned concurrency for critical functions

6. Monitoring & Logging

6.1 CloudWatch

- Lambda logs

- API Gateway access logs
- Custom metrics (water saved, active devices)

6.2 Alarms

- Lambda errors > 5%
- API latency > 3s
- Device offline > 1 hour

7. Disaster Recovery

7.1 Backup Strategy

- DynamoDB: Point-in-time recovery enabled
- S3: Versioning enabled

7.2 Multi-Region (Future)

- DynamoDB Global Tables
- S3 Cross-Region Replication

8. סיכום רכיבים

Service	Resource Name	Purpose
IoT Core	EcoShower-IoT	תקשורת מכשירים
Lambda	EcoShower-* (5 functions)	לוגיקה עסקית
DynamoDB	EcoShower-* (4 tables)	מסד נתונים
API Gateway	EcoShower-API	REST API
Cognito	EcoShower-Users	אימות משתמשים
SNS	EcoShower-Notifications	התראות
S3	ecoshower-frontend	Frontend אחסון
CloudFront	EcoShower-CDN	הפצת תוכן
CloudWatch	EcoShower-Logs	ניטור