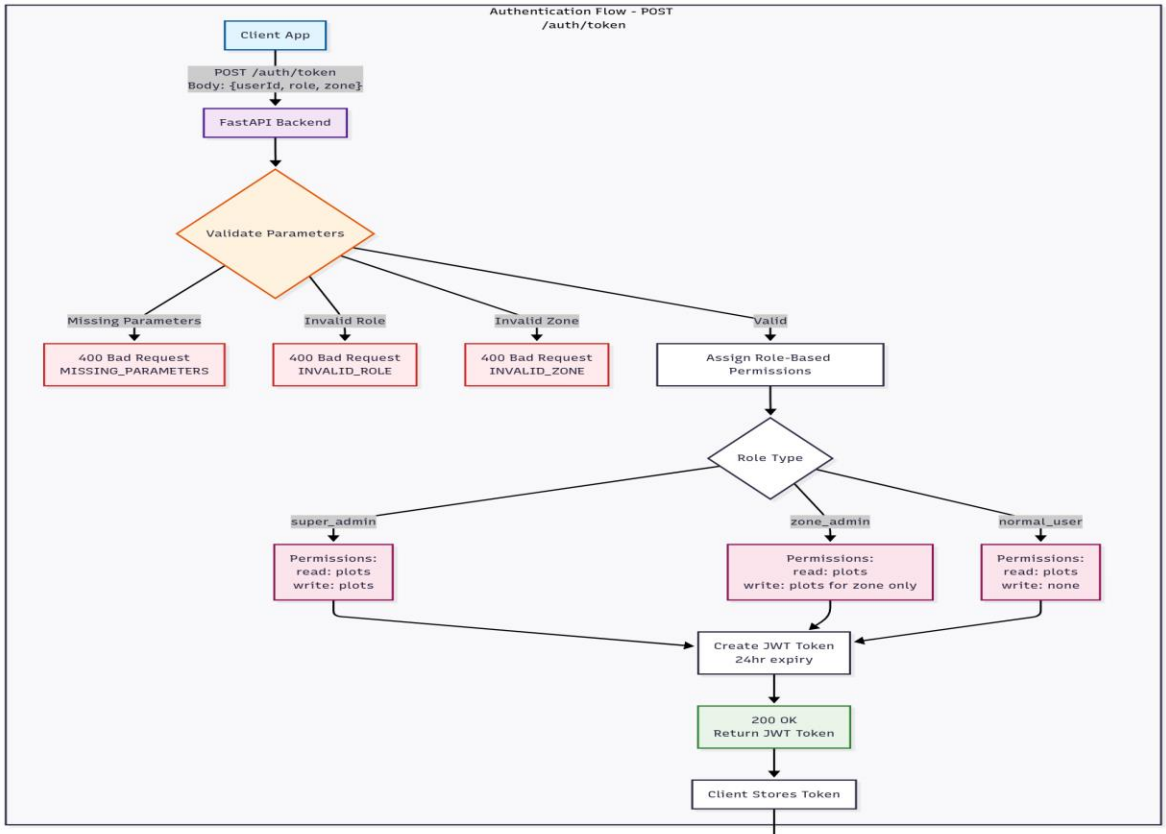
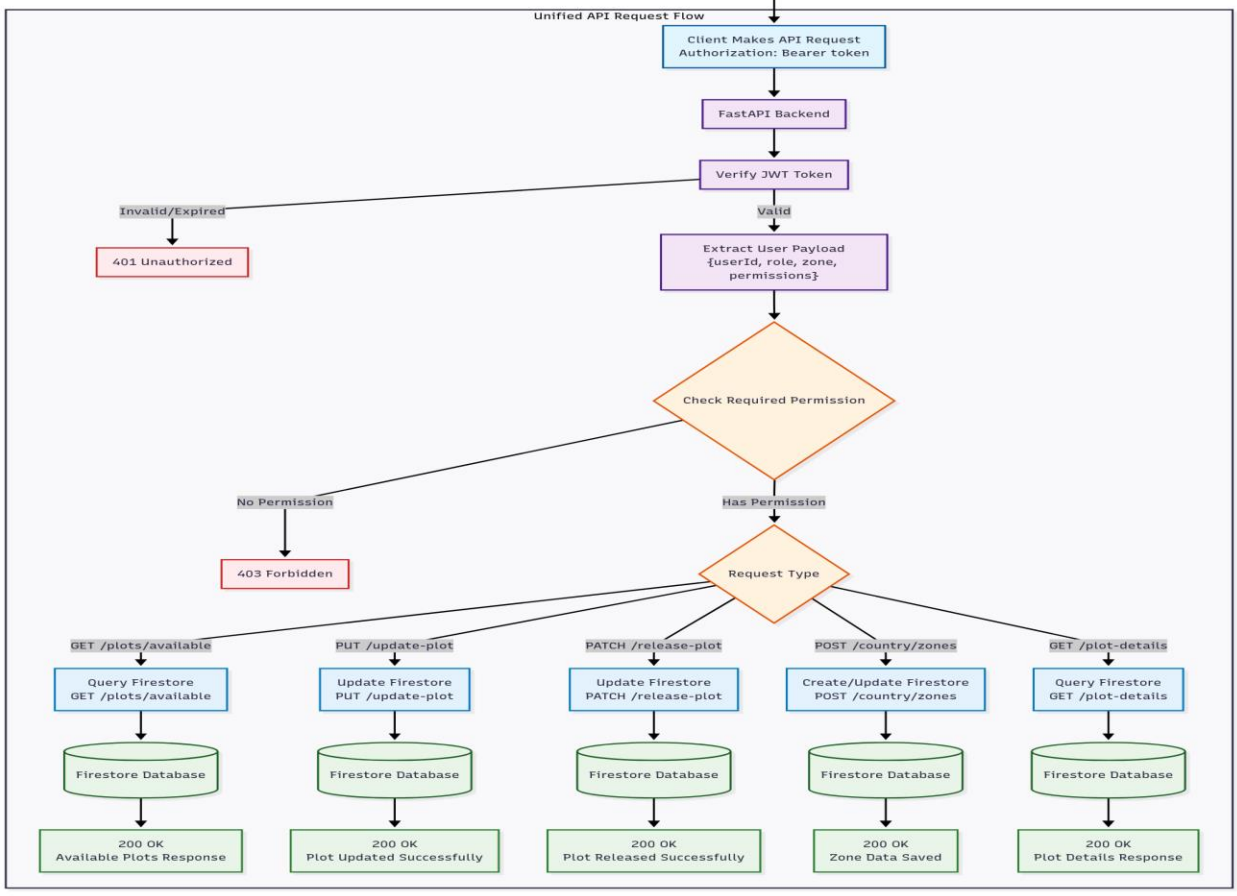
Arise One Platform (AOP) API Documentation


Authentication Flow - POST /auth/token
Client App
POST /auth/token
Body: {userId, role, zone}
FastAPI Backend
Validate Parameters
Missing Parameters
Invalid Role
Invalid Zone
Valid
400 Bad Request
MISSING_PARAMETERS
400 Bad Request
INVALID_ROLE
400 Bad Request
INVALID_ZONE
Assign Role-Based Permissions
Role Type
super_admin
zone_admin
normal_user
Permissions:
read: plots
write: plots
Permissions:
read: plots
write: plots for zone only
Permissions:
read: plots
write: none
Create JWT Token
24hr expiry
200 OK
Return JWT Token
Client Stores Token


Unified API Request Flow
Client Makes API Request
Authorization: Bearer token
FastAPI Backend
Verify JWT Token
Invalid/Expired
401 Unauthorized
Valid
Extract User Payload
{userId, role, zone, permissions}
Check Required Permission
No Permission
403 Forbidden
Has Permission
Request Type
GET /plots/available
PUT /update-plot
PATCH /release-plot
POST /country/zones
GET /plot-details
Query Firestore
GET /plots/available
Update Firestore
PUT /update-plot
Update Firestore
PATCH /release-plot
Create/Update Firestore
POST /country/zones
Query Firestore
GET /plot-details
Firestore Database
Firestore Database
Firestore Database
Firestore Database
Firestore Database
200 OK
Available Plots Response
200 OK
Plot Updated Successfully
200 OK
Plot Released Successfully
200 OK
Zone Data Saved
200 OK
Plot Details Response

1. Auth Api (Authentication Token Generation)

Purpose: Generate JWT token for API access based on user credentials

Method: POST

Endpoint: /auth/token

Request Body (all required):

- userId (string): User identifier
- role (string): User role (super_admin, zone_admin, normal_user)
- zone (string): Zone/Country name

Request Example

```
{
  "userId": "user_001",
  "role": "zone_admin",
  "zone": "Gabon"
}
```

Authorization: None required

Response (200 OK)


```
{
  "token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpYXQiOjE3MTkwMDAwMDAsImV4cCI6MTcxO
TAwMzYwMCwidXNlcklkIjoidXNlcl8wMDEiLCJyb2xlIjoiem9uZV9hZG1pbiIsInpvbmUiOiJHY
WJvbiIsInBlcm1pc3Npb25zIjp7InJlYWQiOlsicGxvdHMiLCJ6b25lcyIsInJlcG9ydHMiXSwid3Jp
dGUiOlsicGxvdHMiLCJ6b25lcyJdfX0.signature_here"
}
```


Response (400 Bad Request)

```
{
  "error": "INVALID_ROLE",
  "message": "Role must be: super_admin, zone_admin, or normal_user"
```

}

Response (400 Bad Request)

{
 "error": "INVALID_ZONE",
 "message": "Invalid zone provided"
}

Response (400 Bad Request)

{
 "error": "MISSING_PARAMETERS",
 "message": "userId, role, and zone are required"
}

Message Elements - Request Body

Element	Data Type	Length	Mandatory/Optional	Description
userId	String	N/A	Mandatory	User identifier
role	String	N/A	Mandatory	User role (super_admin, zone_admin, normal_user)
zone	String	N/A	Mandatory	Zone/Country name

Message Elements - Response (Success)

Element	Data Type	Length	Mandatory/Optional	Description
token	String	N/A	Mandatory	JWT authentication token

Error Responses

Response (400 Bad Request)


```
{
  "error": "INVALID_ROLE",
  "message": "Role must be: super_admin, zone_admin, or normal_user"
}
```


Response (400 Bad Request)


```
{
  "error": "INVALID_ZONE",
  "message": "Invalid zone provided"
}
```


Response (400 Bad Request)


```
{
  "error": "MISSING_PARAMETERS",
  "message": "userId, role, and zone are required"
}
```


1.2 JWT Token Structure

Header


```
{
  "alg": "HS256",
  "typ": "JWT"
}
```


Payload (decoded content)


```
{
  "iat": 1719000000,
```

```
  "exp": 1719003600,
  "userId": "user_001",
  "role": "zone_admin",
  "zone": "Gabon",
  "permissions": {
    "read": ["plots", "zones"],
    "write": ["plots", "zones"]
  }
}
```

Claims Description

- iat: Token issued at time (Unix timestamp)
- exp: Token expiration time (Unix timestamp) - 24 hour from issue
- userId: User identifier from request
- role: User role from request
- zone: Zone/Country from request
- permissions: Auto-assigned based on role

1.3 Role-Based Permissions

super_admin

- read: ["plots"]
- write: ["plots"]

zone_admin

- read: ["plots"]
- write: ["plots"] // for particular zone only

normal_user

- read: ["plots"]
- write: []

1.4. Using Generated Token

For subsequent API calls, include token in Authorization header:

Header Format

Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

Example API Call

GET /plots/available?country=Gabon&zoneCode=GSEZ&category=Residential
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

2. Available Plots Retrieval

Purpose: Populate the "Available Plots" dropdown during customer lead registration.

Method: GET

Endpoint: /plots/available

Query Parameters (all required):

- country (string): country name - Code
- zoneCode (string): Zone identifier
- category (string): Plot category (e.g., Residential, Commercial)
- phase (string): Development phase (currently don't have)

Request Example

/plots/available?country=Gabon&zoneCode=GSEZ&category=Residential&phase=2

Authorization:

Bearer <token>

Response (200 OK)


```
{
 "plots": [
  {
   "plot": "Block A-12",
   "plotStatus": "Available",
   "category": "Residential",
   "phase": "2",
   "areaInSqm": 250.5,
   "areaInHa": 0.025
  }]}
```


Message Elements - Query Parameters

Element	Data Type	Length	Mandatory/Optional	Description
country	String	N/A	Mandatory	Country name/code
zoneCode	String	N/A	Mandatory	Zone identifier
category	String	N/A	Mandatory	Plot category (Residential, Commercial)
phase	String	N/A	Mandatory	Development phase

Message Elements - Response (Success)

Element	Data Type	Length	Mandatory/Optional	Description
plot	String	N/A	Mandatory	Plot identifier
plotStatus	String	N/A	Mandatory	Current plot status
category	String	N/A	Mandatory	Plot category
phase	String	N/A	Mandatory	Development phase
areaInSqm	Number (Float)	N/A	Mandatory	Area in square meters
areaInHa	Number (Float)	N/A	Mandatory	Area in hectares

Error Responses

Response (401 Unauthorized)


```
{
 "error": "UNAUTHORIZED",
 "message": "Invalid or missing authorization token"
```

}

Response (400 Bad Request)

{
 "error": "INVALID_PARAMETERS",
 "message": "Missing required query parameters"
}

Response (404 Not Found)

{
 "error": "NOT_FOUND",
 "message": "No plots found for specified criteria"
}

3. Plot Update

Purpose: Synchronize plot metadata and reflect in UI

Method: PUT

Endpoint: /update-plot

Request Body

{
 "country": "Gabon",
 "zoneCode": "GSEZ",
 "phase": "2",
 "plotName/plotNumber": "Block A-12",
 "companyName": "",
 "sector": "Timber",
 "activity": "furniture",
 "plotStatus": "Occupied",

```
  "operationalStatus": "Active",
  "remark": "East facing plot",   // optional
  "logo": "url"                  // optional
}
```

Authorization:

Bearer <token>

Responses

- 200 OK: { "message": "Plot A-12 updated successfully." }

Message Elements - Request Body

Element	Data Type	Length	Mandatory/Optional	Description
country	String	N/A	Mandatory	Country name
zoneCode	String	N/A	Mandatory	Zone identifier
phase	String	N/A	Mandatory	Development phase
plotName/plotNumber	String	N/A	Mandatory	Plot name or number
companyName	String	N/A	Mandatory	Company name
sector	String	N/A	Mandatory	Business sector
activity	String	N/A	Mandatory	Business activity
plotStatus	String	N/A	Mandatory	Plot status
operationalStatus	String	N/A	Mandatory	Operational status
remark	String	N/A	Optional	Additional remarks
logo	String (URL)	N/A	Optional	Company logo URL

Message Elements - Response (Success)

Element	Data Type	Length	Mandatory/Optional	Description
message	String	N/A	Mandatory	Success confirmation message

Error Responses

Response (401 Unauthorized)


```
{
  "error": "UNAUTHORIZED",
  "message": "Invalid or missing authorization token"
}
```


Response (403 Forbidden)


```
{
  "error": "FORBIDDEN",
  "message": "Insufficient permissions to update plot"
}
```


Response (400 Bad Request)


```
{
  "error": "INVALID_PLOT",
  "message": "Plot not found or invalid plot identifier"
}
```


Response (400 Bad Request)


```
{
  "error": "MISSING_PARAMETERS",
  "message": "Required parameters missing"
}
```


Response (400 Bad Request)

```
{
  "error": "VALIDATION_ERROR",
  "message": "Invalid data format or values"
}
```

4. Plot Release

Purpose: Mark a plot as Available when released

Method: PATCH

Endpoint: /release-plot

Request Body

```
{
  "country": "Benin",
  "zoneCode": "GDIZ",
  "plotStatus": "available",
  "plotNumber/plotName": "Block A-12"
}
```

Authorization:

Bearer <token>

Behavior

- Sets plotStatus to 'Available'

Response

- 200 OK: { "message": "Block Plot A-12 released and marked Available." }

Message Elements - Request Body

Element	Data Type	Length	Mandatory/Optional	Description
country	String	N/A	Mandatory	Country name
zoneCode	String	N/A	Mandatory	Zone identifier
plotStatus	String	N/A	Mandatory	Plot status (should be 'available')
plotNumber/plotName	String	N/A	Mandatory	Plot number or name

Message Elements - Response (Success)

Element	Data Type	Length	Mandatory/Optional	Description
message	String	N/A	Mandatory	Success confirmation message

Error Responses

Response (401 Unauthorized)


```
{
  "error": "UNAUTHORIZED",
  "message": "Invalid or missing authorization token"
}
```


Response (403 Forbidden)


```
{
  "error": "FORBIDDEN",
  "message": "Insufficient permissions to release plot"
}
```


Response (400 Bad Request)

```
{
  "error": "INVALID_PLOT",
  "message": "Plot not found or invalid plot identifier"
}
```


Response (400 Bad Request)


```
{
  "error": "MISSING_PARAMETERS",
  "message": "Required parameters missing"
}
```


Response (400 Bad Request)


```
{
  "error": "INVALID_STATUS",
  "message": "Invalid plot status for release operation"
}
```


5. Zone Master

Purpose: Single source of truth for zone metadata

Method: POST

Endpoint: /country/zones

Request Body:


```
{
  "country": "Gabon",
  "zoneCode": "GSEZ",
  "phase": "1",
```

"landArea": "120 Ha"
}

Responses

- 200 OK:

{
"country": "Gabon",
"zoneCode": "GSEZ",
"phase": "1",
"landArea": "120 Ha"
}

Message Elements - Request Body

Element	Data Type	Length	Mandatory/Optional	Description
country	String	N/A	Mandatory	Country name
zoneCode	String	N/A	Mandatory	Zone identifier
phase	String	N/A	Mandatory	Development phase
landArea	String	N/A	Mandatory	Land area (with unit, e.g., "120 Ha")

Message Elements - Response (Success)

Element	Data Type	Length	Mandatory/Optional	Description
country	String	N/A	Mandatory	Country name
zoneCode	String	N/A	Mandatory	Zone identifier
phase	String	N/A	Mandatory	Development phase
landArea	String	N/A	Mandatory	Land area with unit

Error Responses

Response (401 Unauthorized)

```
{
  "error": "UNAUTHORIZED",
  "message": "Invalid or missing authorization token"
}
```


Response (403 Forbidden)


```
{
  "error": "FORBIDDEN",
  "message": "Insufficient permissions to create/update zone"
}
```


Response (409 Conflict)


```
{
  "error": "DUPLICATE_ZONE",
  "message": "Zone already exists"
}
```


Response (400 Bad Request)


```
{
  "error": "MISSING_PARAMETERS",
  "message": "Required parameters missing"
}
```


Response (400 Bad Request)


```
{
  "error": "VALIDATION_ERROR",
  "message": "Invalid data format or values"
}
```

6. Open API

Purpose: Third-party consumable endpoint

Method: GET

Endpoint: /plot-details?country=Gabon&zoneCode=GSEZ

Query Parameters:

- country (string, required)
- zoneCode (string, required)
- phase (string, optional)
- category (string, optional)

Response (200 OK)


```
{
  "metadata": {
    "country": "Gabon",
    "zoneCode": "GSEZ",
    "phase": 1,
    "category": null
  },
  "plots": [
    {
      "plotName": "Block A-12",
      "category": "Residential",
      "areaInHa": 250.5,
      "areaInSqm": 20.5,
      "sector": "Timber",
      "activity": "furniture",
      "plotStatus": "Available",
      "operationalStatus": "Pending",
      "logo": "example.png",
      "remarks": ""
    }
  ]
```

}

Message Elements - Query Parameters

Element	Data Type	Length	Mandatory/Optional	Description
country	String	N/A	Mandatory	Country name
zoneCode	String	N/A	Mandatory	Zone identifier
phase	String	N/A	Optional	Development phase
category	String	N/A	Optional	Plot category filter

Message Elements - Response (Success)

Element	Data Type	Length	Mandatory/Optional	Description
metadata	Object	N/A	Mandatory	Request metadata
metadata.country	String	N/A	Mandatory	Country name
metadata.zoneCode	String	N/A	Mandatory	Zone identifier
metadata.phase	Number	N/A	Mandatory	Development phase or null
metadata.category	String	N/A	Mandatory	Category filter or null
plots	Array	N/A	Mandatory	Array of plot details
plots[].plotName	String	N/A	Mandatory	Plot name
plots[].category	String	N/A	Mandatory	Plot category
plots[].areaInHa	Number (Float)	N/A	Mandatory	Area in hectares
plots[].areaInSqm	Number (Float)	N/A	Mandatory	Area in square meters
plots[].sector	String	N/A	Mandatory	Business sector
plots[].activity	String	N/A	Mandatory	Business activity
plots[].plotStatus	String	N/A	Mandatory	Current plot status
plots[].operationalStatus	String	N/A	Mandatory	Operational status
plots[].logo	String	N/A	Optional	Company logo filename
plots[].remarks	String	N/A	Optional	Additional remarks

Error Responses

Response (400 Bad Request)


```
{
  "error": "INVALID_PARAMETERS",
  "message": "Missing required query parameters (country, zoneCode)"
```

```
}
```

Response (404 Not Found)

```
{
  "error": "NOT_FOUND",
  "message": "No data found for specified criteria"
}
```

Response (500 Internal Server Error)

```
{
  "error": "INTERNAL_ERROR",
  "message": "Internal server error"
}
```

Common Authorization Header Format

All authenticated endpoints require:

Authorization: Bearer <JWT_TOKEN>

Common HTTP Status Codes

- **200 OK**: Request successful
- **400 Bad Request**: Invalid request parameters or data
- **401 Unauthorized**: Invalid or missing authentication
- **403 Forbidden**: Insufficient permissions
- **404 Not Found**: Resource not found
- **409 Conflict**: Resource already exists
- **500 Internal Server Error**: Server-side error

User Management Endpoints

7. Add New User

Purpose: Create a new user in the system with specified role and zone access

Method: POST

Endpoint: /create_user

Request Body (all required):

```
{
  "email": "gabon@user.com",
  "role": "zone_admin",
  "zone": "Gabon"
}
```

Authorization:

Bearer <token>

Response (201 Created)

```
{
  "message": "User created successfully",
  "user": {
    "email": "john.doe@example.com",
    "role": "zone_admin",
    "zone": "Gabon",
    "createdDate": "2025-07-02T10:30:00Z"
  }
}
```

Response (400 Bad Request)

```
{
  "error": "INVALID_ROLE",
```

```
 "message": "Role must be: super_admin, zone_admin, or normal_user"
}
```


Response (400 Bad Request)


```
{
 "error": "INVALID_ZONE",
 "message": "Invalid zone provided"
}
```


Response (409 Conflict)


```
{
 "error": "USER_EXISTS",
 "message": "User with this email already exists"
}
```


Response (400 Bad Request)


```
{
 "error": "MISSING_PARAMETERS",
 "message": "email, role, and zone are required"
}
```


Message Elements - Request Body

Element	Data Type	Length	Mandatory/Optional	Description
email	String	N/A	Mandatory	User email address (unique identifier)
role	String	N/A	Mandatory	User role (super_admin, zone_admin, normal_user)
zone	String	N/A	Mandatory	Zone/Country assignment

Message Elements - Response (Success)

Element	Data Type	Length	Mandatory/Optional	Description
message	String	N/A	Mandatory	Success confirmation message

user	Object	N/A	Mandatory	Created user details
user.email	String	N/A	Mandatory	User email address
user.role	String	N/A	Mandatory	User role
user.zone	String	N/A	Mandatory	Zone assignment
user.createdDate	String (ISO 8601)	N/A	Mandatory	User creation timestamp

Error Responses

Response (401 Unauthorized)

```
{
  "error": "UNAUTHORIZED",
  "message": "Invalid or missing authorization token"
}
```

Response (403 Forbidden)

```
{
  "error": "FORBIDDEN",
  "message": "Insufficient permissions to create user. Super admin access required."
}
```

Response (400 Bad Request)

```
{
  "error": "INVALID_EMAIL",
  "message": "Invalid email format"
}
```

8. Update User

Purpose: Update existing user role and zone assignment

Method: PUT

Endpoint: /update_user

Request Body:

```
{
  "email": "benin@user.com",
  "role": "normal_user",
  "zone": "Benin"
}
```

Authorization:

Bearer <token>

Response (200 OK)

```
{
  "message": "User benin@user.com updated successfully",
  "user": {
    "email": " benin@user.com",
    "role": "normal_user",
    "zone": "Benin",
    "lastModified": "2025-07-02T14:25:00Z"
  }
}
```

Response (404 Not Found)

```
{
  "error": "USER_NOT_FOUND",
  "message": "User with specified email not found"
}
```

Response (400 Bad Request)

```
{
  "error": "MISSING_EMAIL",
```

"message": "email is required for update operation"
}

Message Elements - Request Body

Element	Data Type	Length	Mandatory/Optional	Description
email	String	N/A	Mandatory	User email address to update
role	String	N/A	Optional	Updated user role
zone	String	N/A	Optional	Updated zone assignment

Message Elements - Response (Success)

Element	Data Type	Length	Mandatory/Optional	Description
message	String	N/A	Mandatory	Success confirmation message
user	Object	N/A	Mandatory	Updated user details
user.email	String	N/A	Mandatory	User email address
user.role	String	N/A	Mandatory	User role
user.zone	String	N/A	Mandatory	Zone assignment
user.lastModified	String (ISO 8601)	N/A	Mandatory	Last modification timestamp

Error Responses

Response (401 Unauthorized)

{
"error": "UNAUTHORIZED",
"message": "Invalid or missing authorization token"
}

Response (403 Forbidden)

{
"error": "FORBIDDEN",
"message": "Insufficient permissions to update this user"
}

Response (400 Bad Request)


```
{
  "error": "INVALID_ROLE",
  "message": "Role must be: super_admin, zone_admin, or normal_user"
}
```


Response (400 Bad Request)


```
{
  "error": "INVALID_ZONE",
  "message": "Invalid zone provided"
}
```


Response (400 Bad Request)


```
{
  "error": "INVALID_EMAIL",
  "message": "Invalid email format"
}
```


Response (400 Bad Request)


```
{
  "error": "VALIDATION_ERROR",
  "message": "Invalid data format or values"
}
```