

# ACME CORP

## LED Name Badge API

**Integration Guide & Reference Manual**

Version: 1.0.0 | January 2026

## 1. Overview

---

Welcome to the Acme LED Name Badge API. This interface allows developers and integrators to programmatically control the display of our connected LED badges. This manual outlines the available endpoints, required parameters, and standard response formats.

### Base Configuration

By default, the API service listens on the following local address:

```
Base URL: http://localhost:5001  
Content-Type: application/json
```

## 2. API Endpoints

---

### 2.1 Update Badge Display

Updates the text and visual content currently shown on the LED name badge. This endpoint accepts alphanumeric text and supports icon codes.

```
POST /display-text
```

#### Example Request:

```
curl -X POST http://localhost:5001/display-text \  
-H "Content-Type: application/json" \  
-d '{"text": "Hello World! {icon_smile}"}
```

### 2.2 Retrieve Icon Library

Retrieves a list of all predefined icons stored in the device firmware.

```
GET /predefined-icons
```

#### Example Response:

```
[ "icon_smile", "icon_heart", "icon_coffee", "icon_star" ]
```

## 3. Error Handling

---

The API utilizes standard HTTP status codes:

200 OK: Request processed successfully.  
400 Bad Request: Malformed request (missing 'text').  
500 Internal Error: Hardware/Server failure.

## 4. Agent Tool Definitions

Use the following definitions for LLM or Agent integration.

### Function: display\_text

```
{  
  "name": "display_text",  
  "description": "Display text on the LED name badge.",  
  "parameters": {  
    "type": "object",  
    "properties": {  
      "text": {  
        "type": "string",  
        "description": "The text to display..."  
      }  
    },  
    "required": ["text"]  
  }  
}
```

### Function: get\_predefined\_icons

```
{  
  "name": "get_predefined_icons",  
  "description": "Retrieve predefined icons...",  
  "parameters": {  
    "type": "object",  
    "properties": {},  
    "required": []  
  }  
}
```