# Detailed Documentation on Logging User Interactions in React/JavaScript Applications

## Overview

This document outlines the structure, best practices, and examples for logging user interactions in React/JavaScript applications. The goal is to ensure that the system can:  
- Monitor user interactions.  
- Troubleshoot issues efficiently.  
- Gain insights into user behavior through well-structured logs.

## 1. Where Are We Logging?

### Logging Destination:

Logs are sent to a centralized logging platform, such as ElasticSearch or Kibana, for aggregation, visualization, and querying.

### URLs & Settings:

- Logging Endpoint: `https://logs.example.com/ingest`  
- Configuration Settings:  
 - Log Level: Configured via environment variables (`DEBUG`, `INFO`, `ERROR`).  
 - Batch Logging: Logs are batched and sent every 5 seconds to reduce network overhead.  
 - Retention Policy: Logs are retained for 90 days for analysis.

## 2. What Are We Logging in Case of UI Actions/Events?

Fields Logged for UI Actions:  
| Field | Description |  
|----------------|-------------|  
| @timestamp | Time the event occurred. |  
| event\_type | Always `USER\_ACTION` for UI events. |  
| action | Type of user action (e.g., `click`, `select`, `submit`). |  
| user\_id | Identifier for the user performing the action. |  
| session\_id | Unique session identifier. |  
| page\_url | URL of the page where the action occurred. |  
| element\_id | ID or unique identifier of the UI element involved. |  
| details | Additional details such as selected options or form input (anonymized). |  
| metadata | Environment details (e.g., `production`, `1.0.0`, `web`). |

### Example Log (Button Click):

{  
 "@timestamp": "2024-12-25T10:05:00Z",  
 "event\_type": "USER\_ACTION",  
 "action": "click",  
 "context": {  
 "user": {  
 "user\_id": "12345",  
 "session\_id": "abcde12345"  
 },  
 "element": {  
 "id": "submit-button",  
 "type": "button",  
 "action": "click"  
 },  
 "page\_url": "/signup"  
 },  
 "metadata": {  
 "environment": "production"  
 }  
}

## 3. What Are We Logging in Case of API Calls?

Fields Logged for API Calls:  
| Field | Description |  
|----------------|-------------|  
| @timestamp | Time the API call was made. |  
| event\_type | Always `API\_CALL`. |  
| action | Type of API action (e.g., `fetch`, `submit`). |  
| request\_url | Full endpoint of the API being called. |  
| method | HTTP method used (`GET`, `POST`, etc.). |  
| status\_code | HTTP response status code. |  
| response\_time\_ms | Time taken to receive a response. |  
| error\_details | Error information in case of failure. |

### Example Log (Success):

{  
 "@timestamp": "2024-12-25T10:01:00Z",  
 "event\_type": "API\_CALL",  
 "action": "fetch",  
 "context": {  
 "request\_url": "/api/v1/orders",  
 "method": "POST",  
 "status\_code": 200,  
 "response\_time\_ms": 150  
 },  
 "metadata": {  
 "environment": "production",  
 "application\_version": "1.0.0"  
 }  
}