

Mobile application tracker

```
#include <stdio.h>

#include <stdlib.h>

#include <string.h>

// SDK Module – User Actions

Typedef struct SDKNode {

    Char action[100];

    Char timestamp[50];

    Struct SDKNode* next;

} SDKNode;

// Auth Module – Users (logged-in or guest)

Typedef struct AuthNode {

    Char user_id[50];

    Int is_logged_in; // 1 = logged in, 0 = guest

    Struct AuthNode* next;

} AuthNode;

// Event Logger – Events

Typedef struct EventNode {

    Char event_name[100];

    Char timestamp[50];

    Struct EventNode* next;

} EventNode

// Session Tracker – Sessions

Typedef struct SessionNode {
```

```

    Char session_id[50];
    Char start_time[50];
    Char end_time[50];
    Struct SessionNode* next;
} SessionNode;

// Geo Tracker – Geo points
Typedef struct GeoNode {
    Double latitude;
    Double longitude;
    Char timestamp[50];
    Struct GeoNode* next;
} GeoNode;

// Insert SDK action into linked list
Void insertSDKAction(SDKNode** head, const char* action, const char*
timestamp) {
    SDKNode* new_node = (SDKNode*)malloc(sizeof(SDKNode));
    Strcpy(new_node->action, action);
    Strcpy(new_node->timestamp, timestamp);
    New_node->next = *head;
    *head = new_node;
}

// Insert Auth user into linked list
Void insertAuthUser(AuthNode** head, const char* user_id, int is_logged_in) {
    AuthNode* new_node = (AuthNode*)malloc(sizeof(AuthNode));
    Strcpy(new_node->user_id, user_id);

```

```

New_node->is_logged_in = is_logged_in;

New_node->next = *head;

*head = new_node;

}

// Insert Event into linked list

Void insertEvent(EventNode** head, const char* event_name, const char*
timestamp) {

    EventNode* new_node = (EventNode*)malloc(sizeof(EventNode));

    Strcpy(new_node->event_name, event_name);

    Strcpy(new_node->timestamp, timestamp);

    New_node->next = *head;

    *head = new_node;

}

// Insert Session into linked list

Void insertSession(SessionNode** head, const char* session_id, const char*
start_time, const char* end_time) {

    SessionNode* new_node = (SessionNode*)malloc(sizeof(SessionNode));

    Strcpy(new_node->session_id, session_id);

    Strcpy(new_node->start_time, start_time);

    Strcpy(new_node->end_time, end_time);

    New_node->next = *head;

    *head = new_node;

}

// Insert Geo point into linked list

```

```
Void insertGeoPoint(GeoNode** head, double latitude, double longitude, const  
char* timestamp) {
```

```
    GeoNode* new_node = (GeoNode*)malloc(sizeof(GeoNode));
```

```
    New_node->latitude = latitude;
```

```
    New_node->longitude = longitude;
```

```
    Strcpy(new_node->timestamp, timestamp);
```

```
    New_node->next = *head;
```

```
    *head = new_node;
```

```
}
```

```
// Print functions for each module
```

```
Void printSDKActions(SDKNode* head) {
```

```
    Printf("SDK Actions:\n");
```

```
    While (head) {
```

```
        Printf("Action: %s, Time: %s\n", head->action, head->timestamp);
```

```
        Head = head->next;
```

```
    }
```

```
}
```

```
Void printAuthUsers(AuthNode* head) {
```

```
    Printf("Auth Users:\n");
```

```
    While (head) {
```

```
        Printf("User ID: %s, Logged In: %s\n", head->user_id, head->is_logged_in ?  
"Yes" : "Guest");
```

```
        Head = head->next;
```

```
    }
```

```
}
```

```

Void printEvents(EventNode* head) {
    Printf("Events:\n");
    While (head) {
        Printf("Event: %s, Time: %s\n", head->event_name, head->timestamp);
        Head = head->next;
    }
}

Void printSessions(SessionNode* head) {
    Printf("Sessions:\n");
    While (head) {
        Printf("Session ID: %s, Start: %s, End: %s\n", head->session_id, head->start_time, head->end_time);
        Head = head->next;
    }
}

Void printGeoPoints(GeoNode* head) {
    Printf("Geo Points:\n");
    While (head) {
        Printf("Lat: %.4f, Long: %.4f, Time: %s\n", head->latitude, head->longitude, head->timestamp);
        Head = head->next;
    }
}

Int main() {
    SDKNode* sdk_actions = NULL;

```

```
AuthNode* auth_users = NULL;
EventNode* events = NULL;
SessionNode* sessions = NULL;
GeoNode* geo_points = NULL;
// Insert sample data
insertSDKAction(&sdk_actions, "SwipeLeft", "2025-05-24T12:01:00Z");
insertSDKAction(&sdk_actions, "TapButton", "2025-05-24T12:02:00Z");
insertAuthUser(&auth_users, "user123", 1);
insertAuthUser(&auth_users, "guest001", 0);
insertEvent(&events, "PageView", "2025-05-24T12:03:00Z");
insertEvent(&events, "ButtonClick", "2025-05-24T12:04:00Z");
insertSession(&sessions, "session001", "12:00", "12:30");
insertSession(&sessions, "session002", "12:31", "13:00");
insertGeoPoint(&geo_points, 37.7749, -122.4194, "2025-05-24T12:00:00Z");
insertGeoPoint(&geo_points, 37.7750, -122.4195, "2025-05-24T12:05:00Z");
// Print all
printSDKActions(sdk_actions);
printAuthUsers(auth_users);
printEvents(events);
printSessions(sessions);
printGeoPoints(geo_points);
return 0;
}
```