Application Description

madDiscovery is for use by people attending various types of public events (i.e., festivals, community events, etc.). The app will allow the user to enter, store and upload reports of the event as it is happening.

SECTION 1

Feature	Implementation Level
a) Basic Details Input Screen	• Fully Implemented
b) Form Validation	• Partially Implemented in Hybrid Application & native application.
c) Store, View & Delete	• Implemented in native Version but I wasn't fully able to connect my Hybrid version to the database.
d) Search	• I wasn't also able to implement an event search function.
e) Add Report Input Screen	• Fully Implemented in both versions

SECTION 2

- The Hybrid Version fails to display entered location, date & time.
- Database table failed to create when called.
- Device could not connect to SQLite Cordova Plugin installed.
- The Android Native Application does nothing on user clicking the search icon.
- The search icon on the hybrid application failed to appear
- Users have to continuously click the back button on their device to exit the toast screen.

SECTION 3

The user interface of the native (Android) madDiscovery was properly designed with differently styled buttons and backgrounds to keep the users engaged.

Th user interface of the hybrid version was keep minimalistic to ensure the user does not have to navigate continuously through the application. Updating and deleting an event could be achieved by entering the event ID (1…n). This was in an effort to maintain a single page event entry.

APPLICATION LIMITATIONS

- Search functionality missing in both native and PhoneGap versions.
- Delete function missing in native application
- PhoneGap failed to display user entered event report and organizers.

SECTION 4

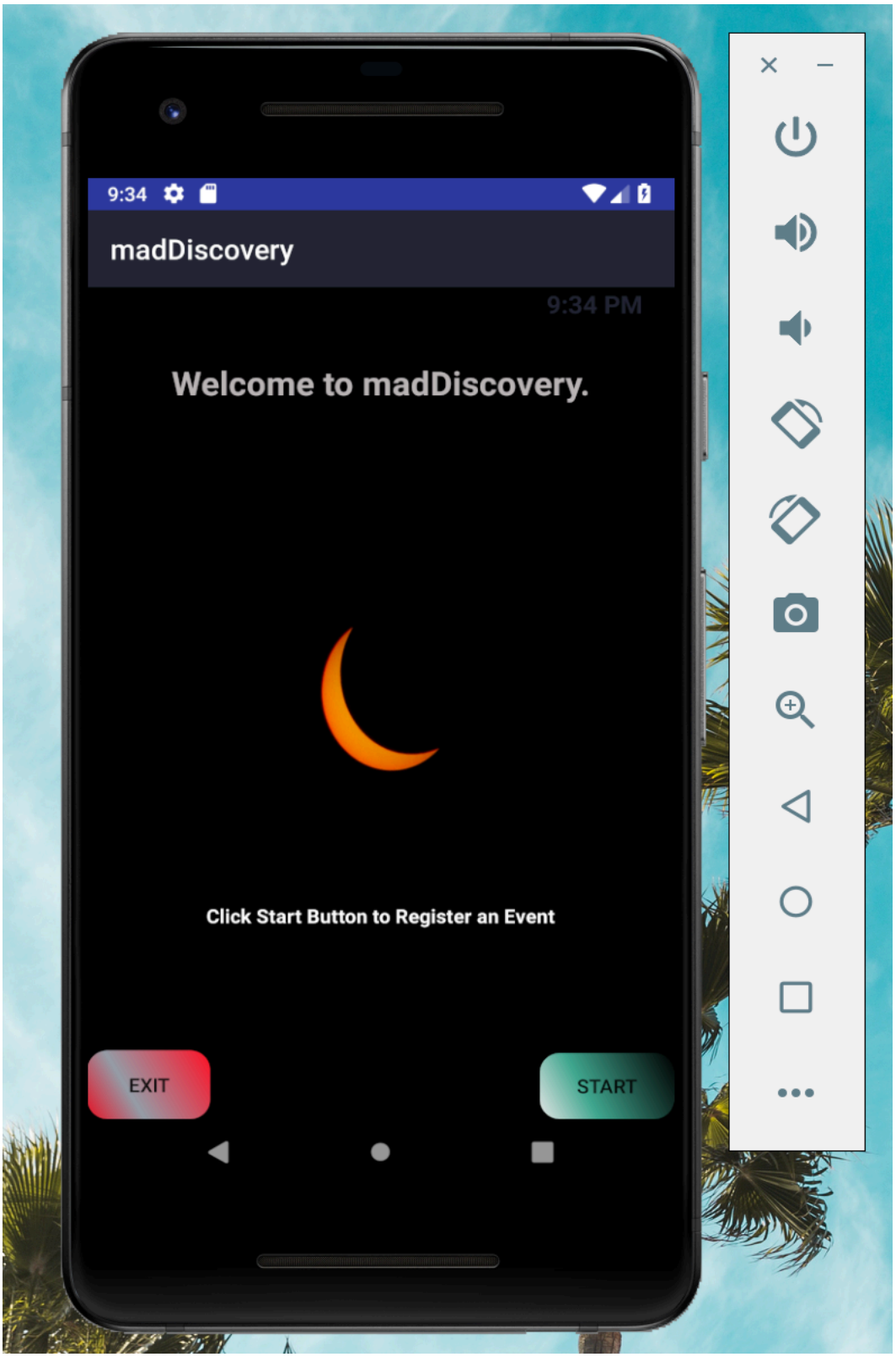

Figure 1 shows welcome screen for native android app

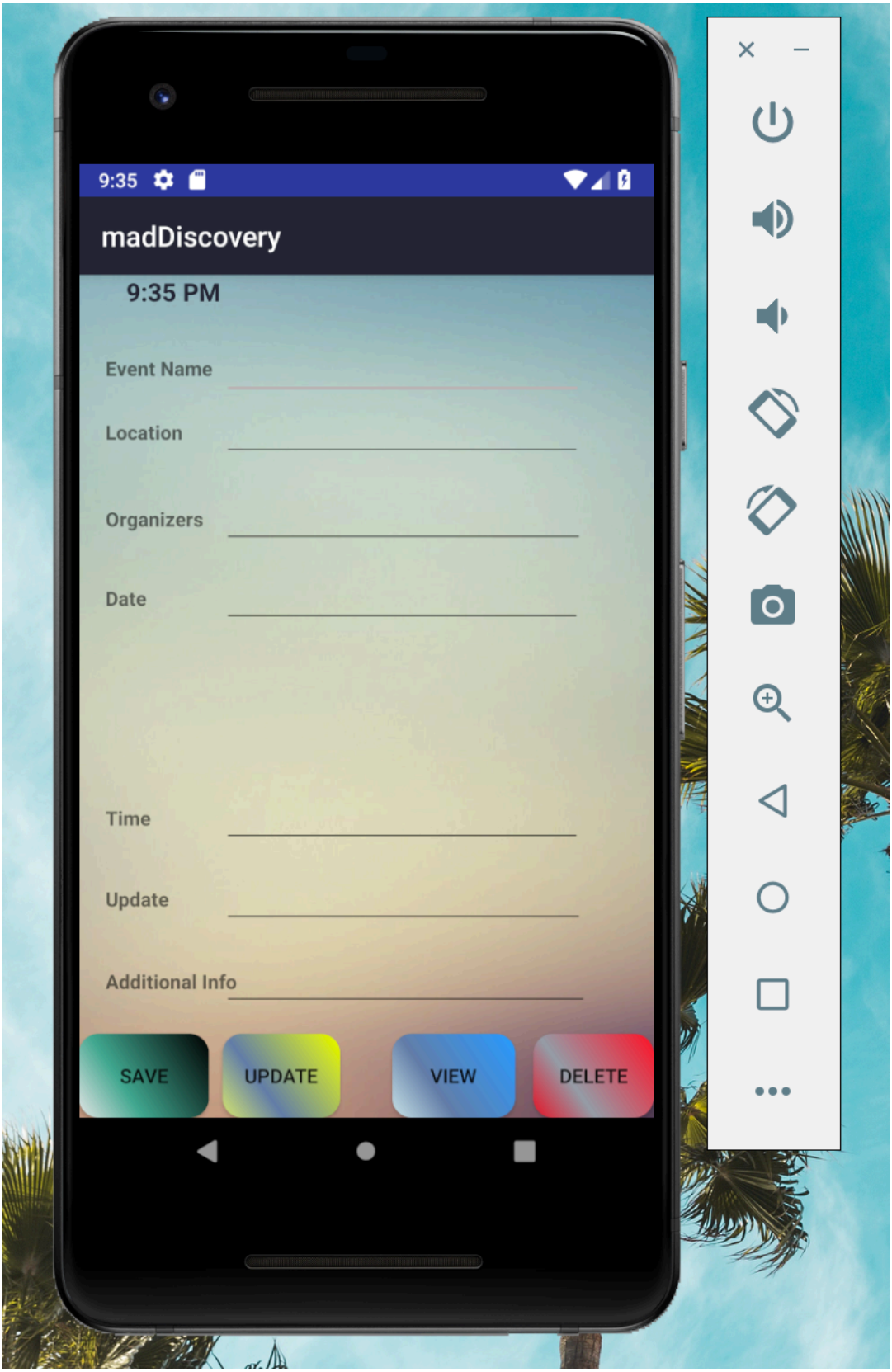

Figure 2 shows event registration screen on native app

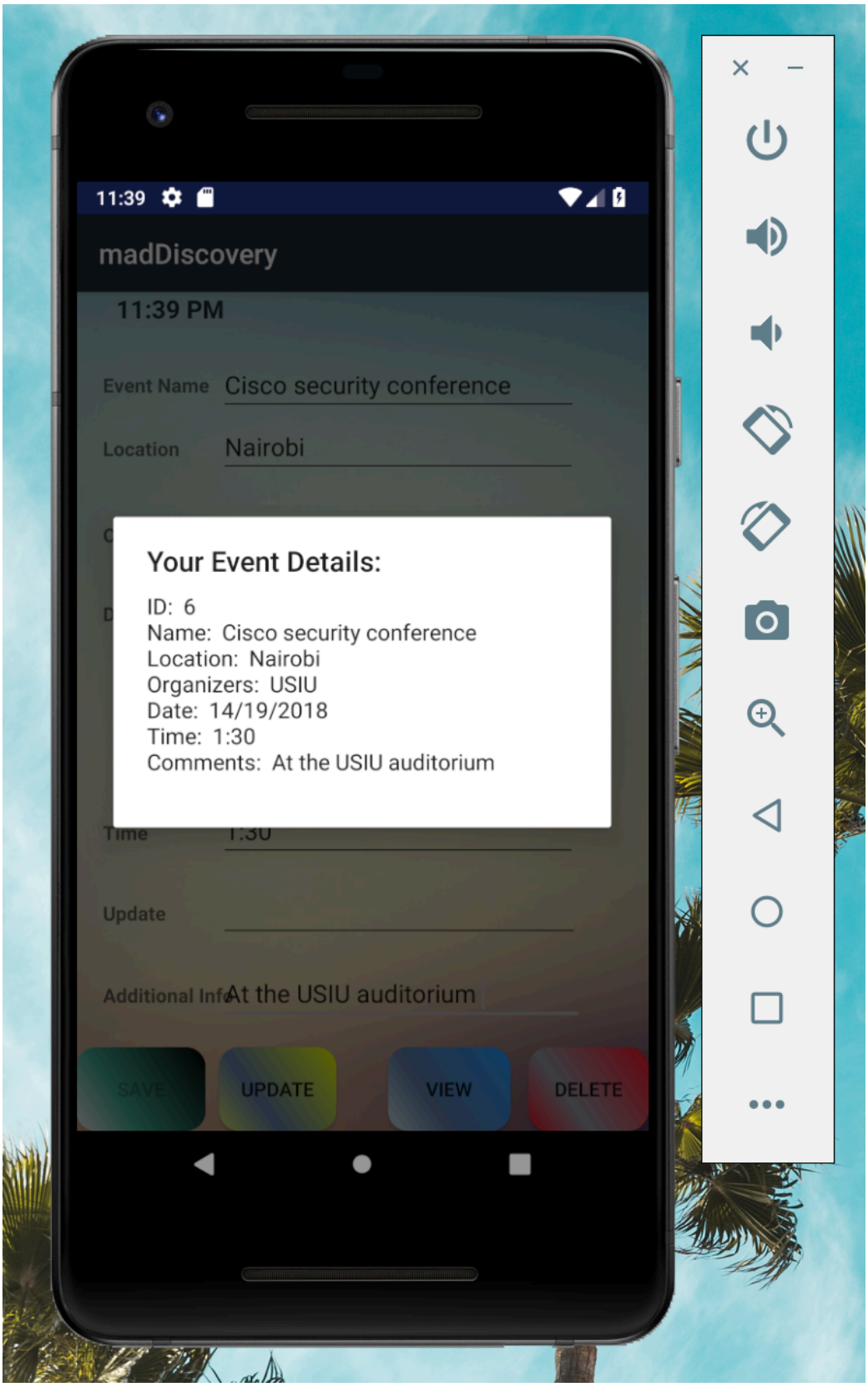

Figure 3 on user click view, event are displayed in a toast

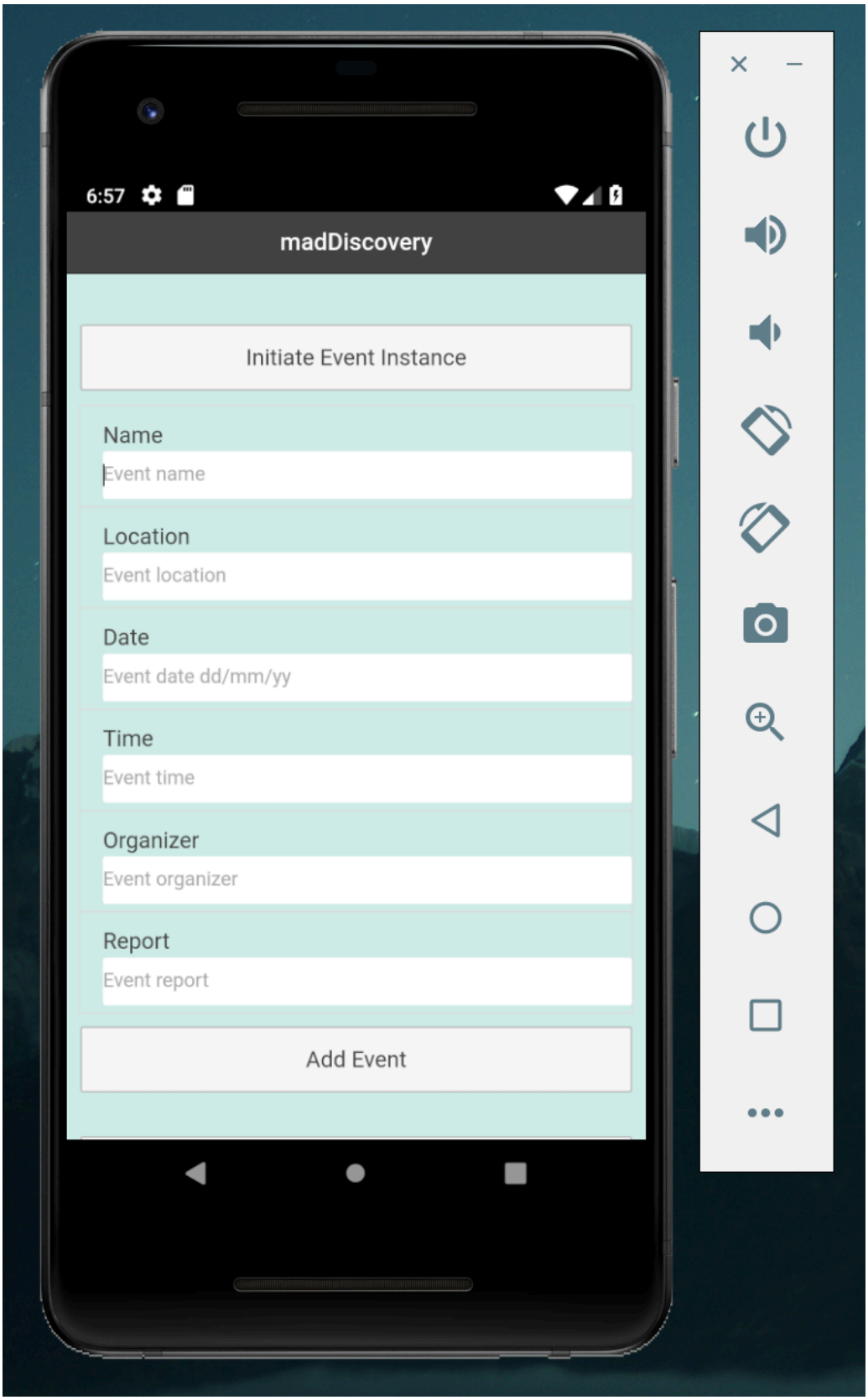

Figure 4 event registration screen for phonegap app

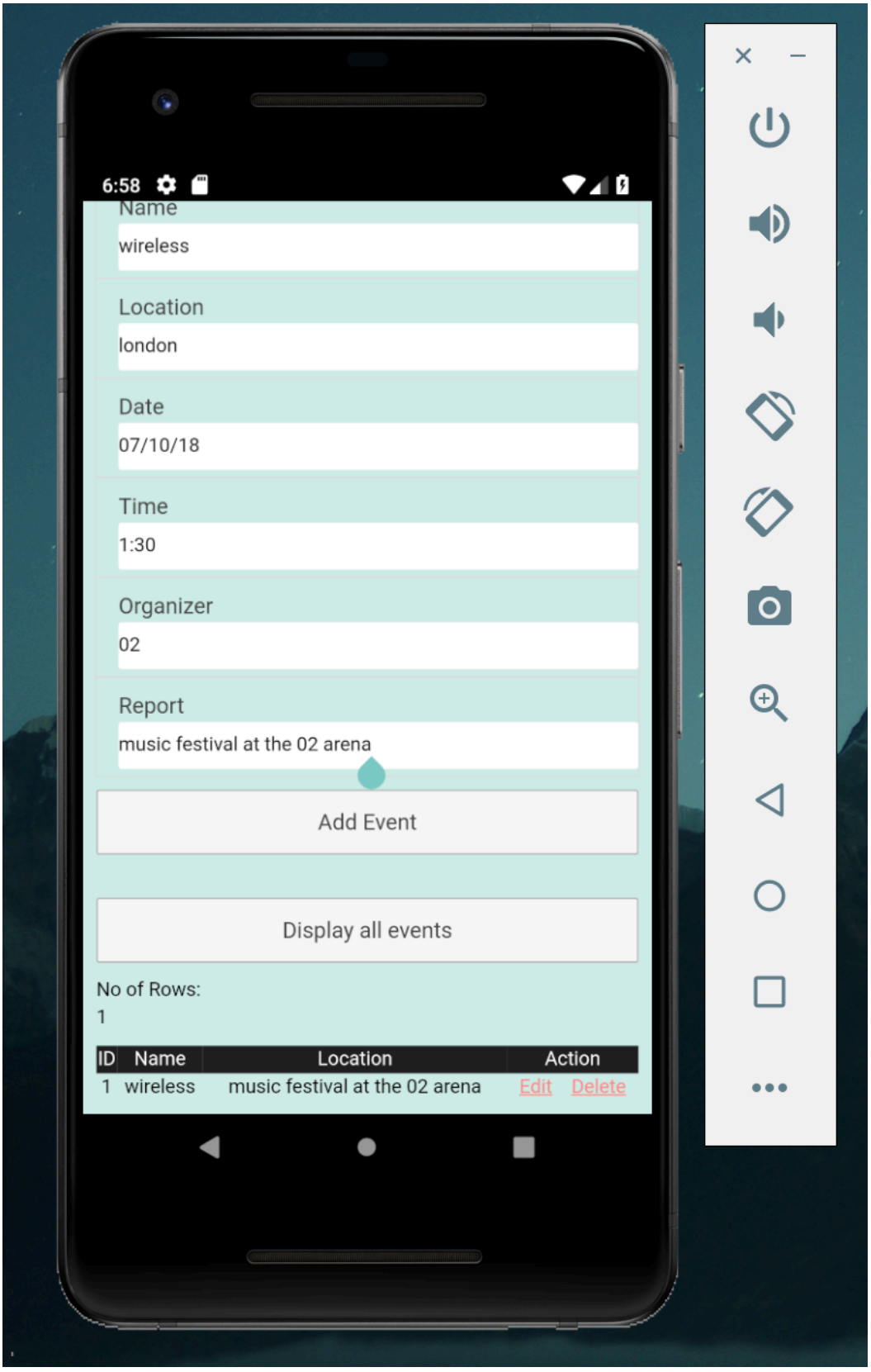

Figure 5 shows the registered events when user clicks "display all events".

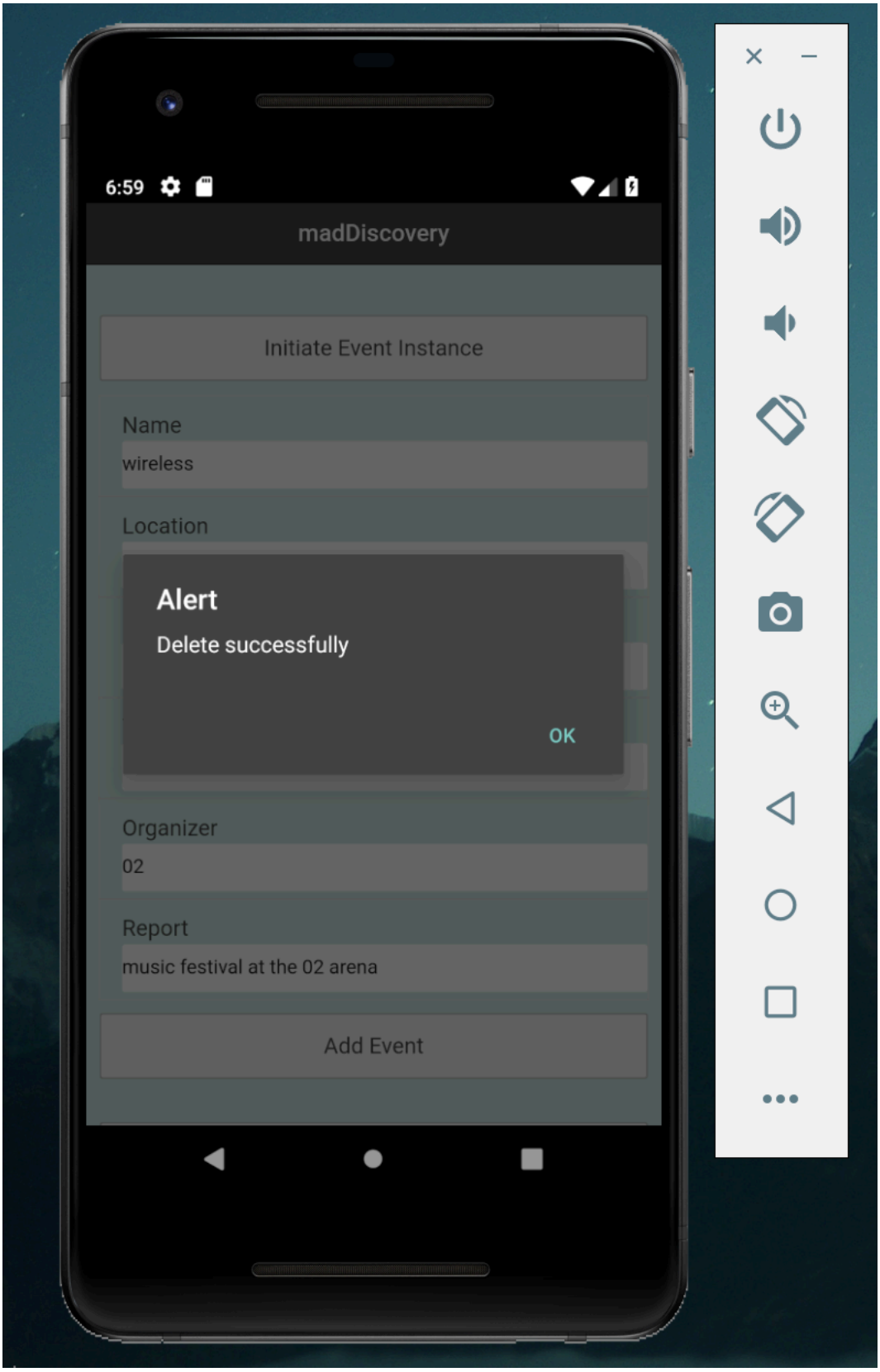

Figure 6 shows the successful deletion of an event entry.

SECTION 5

My madDiscovery for native Android tried to provide the very basic functionalities detailed in the coursework requirement which were store, view and delete event instances. Additionally, I tried to implement an SQLite Plugin in both versions of the application to store the user defined event details. However, I only succeeded in doing so on the native android application.

For a live deployment of madDiscovery, both in the native and hybrid environment will require the restructuring of application files and codes to follow standard set by application hosts such as Google Play Store and Apple App Store.

Security standards for application development were not followed and the applications could be susceptible to sql injection attacks. The application was keep minimalistic to create a human-computer interaction that was also efficient for less experienced users.

Developing madDiscovery both as a native android and a hybrid application made use of already popularly known programming languages including java, JavaScript, CSS and html. This ensure the easy maintenance of the code in future application use.

madDiscovery might sometimes pose a challenge to use for event goers as they are not able to add multiple reports for the same event. Realistically, an event application such as madDiscovery should afford the user the option to quickly edit their entry option as they deem fit before the event details are stored on the SQL database. I was not able to implement this feature in both.

I imported a number of SQL Plugins from open source projects on GitHub as well as officially website of Cordova, PhoneGap.

REFERENCES TO EXTERNAL CODE USED IN APPLICATION DEVELOPMENT

https://developer.android.com
https://cordova.apache.org
https://github.com/litehelpers/Cordova-sqlite-storage/tree/storage-master/www

https://build.phonegap.com/apps
https://www.tutorialspoint.com/cordova/cordova_config_xml.htm
https://www.npmjs.com/package/cordova-sqlite-plugin
http://docs.telerik.com/platform/samples/Sample-To-Do-App-with-SQLite/
https://github.com/mitchtabian/SQLiteSaveUserData/tree/master/app/src/main/java/practice/application/sqlitesaveuserdata