**Lyle Simpson: S1436436**

**Mobile Platform Development – Documented Testing**

|  |  |  |
| --- | --- | --- |
| Test | Expected Result | Actual Result |
| Does the earthquake list activity load when the “Start Application” button is clicked? | A list of all earthquakes, gathered from the API and parsed accordingly. | The correct list was shown |
| What happens when an individual earthquake on the list-view is clicked/pressed? | A detail view should load up for that earthquake clicked, complete with a map of the earthquake’s location. | The detail view was correctly shown, with the map showing the location of the earthquake correctly. |
| Does the application’s activities show sensible and aesthetic layouts in both portrait and orientation mode? | The layouts should be aesthetically appealing in both landscape and portrait mode. Different layouts should be defined where necessary for these orientations | The application has different layouts for landscape and portrait mode on the landing page and the detail view page. With the exception of the statistics view page, all pages show well in both landscape and portrait. The statistics activity has a slight problem in landscape with text disappearing a little at the bottom, and could be improved. |
| Does the application successfully show maps displaying earthquakes locations? | The relevant activities should display a GoogleMap object in their screen. | After initially not showing due to not having an API key, an API key was acquired and added to the AndroidManifest.xml file. The maps then showed properly on the activities that used them, and worked correctly. |
| What happens when the marker representing an earthquake’s location on a map is clicked? | A tooltip should pop-up showing the earthquake’s location by name, and the date the earthquake occurred. | A tooltip displayed near the earthquake’s marker, correctly displaying additional information about the earthquake. |
| When the user enters a search query to the list-view (example: searching for “surrey”), does the list filter accordingly? | When the user enters “SURREY” – only earthquakes whose location contains that substring (regardless of case) should show. | The list filters correctly, showing only earthquakes in Surrey – this works whether entered in lowercase, uppercase or a mix. |
| When the user enters a date using the datepicker, does the list filter correctly, or show a message if no earthquakes exist on that date? | The user enters a date, and only earthquakes on that particular date should be shown in the resulting list. If not earthquakes are found on that date, a message is displayed. | The list filters correctly, only showing earthquakes for a particular date. A message is displayed via Android’s Toast class if no earthquakes occurred on that date. |
| When the user orders the list by the date of the earthquakes, does the list respond accordingly? | If “most recent” is selected, the most recent earthquakes should show at the top. If “least recent”, then the least recent should show at the top | The list is ordered correctly according to each earthquake’s date after choosing either of these options. |
| When the user orders the list by the location (name) of the earthquakes, does the list respond accordingly? | If “A-Z” is selected, the list is ordered alphabetically by location name. If “Z-A” is selected, the list is in descending order of location name. | The list is ordered correctly according to each earthquake’s location name after choosing either of these options. |
| When the user orders the list by the magnitude of the earthquakes, does the list respond accordingly? | If “Highest-Lowest” is selected, then the list is ordered by the largest magnitude earthquake first. If “Lowest-Highest”, then the lowest magnitude earthquake is first in the list. | The list is ordered correctly according to each earthquake’s magnitude after choosing either of these options. |
| When the user orders the list by the depth of the earthquakes, does the list respond accordingly? | If “Highest-Lowest” is selected, then the list is ordered by the deepest earthquake first. If “Lowest-Highest”, then the lowest depth earthquake is first in the list. | The list is ordered correctly according to each earthquake’s depth after choosing either of these options. |
| When the user orders the list by the geographical position of the earthquakes, does the list respond accordingly? | If “Most Northern” is selected, the most northerly earthquakes are shown first. “Most Southern” shows most southerly first, “Most Western” most westerly, and “Most Eastern” most easterly. | The list is ordered correctly according to each earthquake’s position after choosing any of these four options. |
| When any of the navigation items at the bottom of each activity are clicked, does the correct activity get displayed after the navigation event? | There are 3 navigation items: “List” shown navigate to the list view, “Map” to the map view, and “Stats” to the stats view. | Each activity is correctly shown when the navigation items are clicked. This works for all navigation items, on all activities. |
| Does the XML parsing class correctly parse the XML String data into domain objects? | The XML API data should be processed by the XmlPullParser in order to create Earthquake, Location and Coordinates objects for use in the application. | The XML data is correctly parsed, and the attributes of the Earthquake, Location and Coordinates objects are correctly populated via setter methods after the data is parsed |
| Does the application continue to operate when there is no network connectivity? | The application should still be navigable when offline | The application can still be used normally when offline. However, if the user tries to initiate a new API request for new data, the app will become unresponsive. There is scope for handling this aspect of the offline experience better. |
| Is the user able to refresh the list of earthquakes from the API? | The list should be re-generated after the network call is complete | The list is correctly regenerated and refreshed upon initiation by the user. |
| Does the network request, and the request by the user to refresh the data, occur on a separate thread? | The HTTP request should be created on a separate thread from the main application, for performance reasons. | The web request is performed by a Callable class, and is executed on a thread created by Java’s ExecuterService. This successfully prevents the main thread being blocked by the network call. |
| Does the application separate concerns using the MVVM pattern? | The application’s activities should not be overloaded with business logic – this should be moved to viewmodel and repository classes. | The application successfully uses the Repository pattern to provide a wrapper around the data sources (the database, and the network API). The ViewModel mediates between the repository and the activities. The activities are only concerned with processing user data and rendering UI elements. This clean separation of concerns helps the app’s maintainability. |
| Does the application periodically update the list of earthquakes every 5 minutes? | The application should have a Timer in place that automatically performs the web request to the BGS API every 5 minutes. This should automatically refresh the list. | The application successfully utilizes an asynchronous timer which invokes the Callable web request on a separate thread. Every 5 minutes, the web request is made. This is confirmed by the logcat debug output. |