

Contents lists available at [ScienceDirect](http://ScienceDirect)

SoftwareX

journal homepage: [www.elsevier.com/locate/softx](http://www.elsevier.com/locate/softx)

Original software publication

## MobiQ: A modular Android application for collecting social interaction, repeated survey, GPS and photographic data



Suleiman Y. Yerima <sup>a,\*</sup>, Michael Loughlin <sup>b</sup>, Sakir Sezer <sup>b</sup>, John Moriarty <sup>c</sup>, Mark McCann <sup>d</sup>, Helen McAneney <sup>e</sup>, Leeanne O'Hara <sup>c</sup>, Mark A. Tully <sup>f</sup>, Paul S. Ell <sup>g</sup>, Robert Miller <sup>c</sup>, Geraldine Macdonald <sup>c</sup>

<sup>a</sup> Cyber Security Centre, Cyber Technology Institute, De Montfort University, Leicester, LE1 9BH, England, UK

<sup>b</sup> Centre for Secure Information Technologies, Queen's University Belfast, Belfast, Northern Ireland, BT3 9DT, UK

<sup>c</sup> School of Sociology, Social Policy and Social Work/Institute of Child Care Research, Queen's University Belfast, Northern Ireland, UK

<sup>d</sup> MRC/CSO Social and Public Health Sciences Unit, University of Glasgow, 200 Renfield Street, Glasgow, G2 3QB, UK

<sup>e</sup> Centre for Public Health, Queen's University Belfast, Belfast, Northern Ireland, BT12 6BJ, UK

<sup>f</sup> UKCRC Centre of Excellence for Public Health, School of Medicine, Dentistry and Biomedical Science, Queen's University Belfast, Northern Ireland, UK

<sup>g</sup> Centre for Data Digitisation and Analysis, School of Geography, Archaeology and Palaeoecology, Queen's University Belfast, Northern Ireland, UK

### ARTICLE INFO

#### Article history:

Received 14 January 2016

Received in revised form 19 April 2018

Accepted 26 April 2018

#### Keywords:

Longitudinal surveys

Android

Social interaction

GPS

Behaviour research

Social research

Smartphone-based data collection

GIS

e-research

### ABSTRACT

The MobiQ app for Android smartphones is a feature-rich application enabling a novel approach to data collection for longitudinal surveys. It combines continuous automatic background data collection with user supplied data. It can prompt users to complete questionnaires at regular intervals, and allows users to upload photographs for social research projects. The app has the capability to collect GPS location data, and calls and text frequency (excluding content) unobtrusively. The app transmits data to a secure cloud rather than storing research data on the phone, but can also store data temporarily if a data connection is unavailable; hence, MobiQ offers data security advantages over text- or web-based surveys using phones. MobiQ has been pilot tested in the field in a social science research project and is able to collect longitudinal social research data. Due to its modular and flexible design, MobiQ can easily be adapted to suit different research questions. Furthermore, its core design approach which allows for long-term power efficient data collection can be re-used outside the social sciences domain for other kinds of smartphone-based data-driven projects. Projects that have a requirement for communications-based, sensors-based, user-based data collection or any combination of these may find our code and design approach beneficial. For example, MobiQ code and architecture has been successfully adapted to build an app for a project investigating smartphone-based implicit authentication for mobile access control.

© 2018 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

### Code metadata

Current code version	v 0.3
Permanent link to code/repository used of this code version	<a href="https://github.com/ElsevierSoftwareX/SOFTX-D-16-00014">https://github.com/ElsevierSoftwareX/SOFTX-D-16-00014</a>
Legal Code Licence	GPLv2
Code versioning system used	git
Software code languages, tools, and services used	Java, Android SDK, Eclipse IDE
Compilation requirements, operating environments & dependencies	Eclipse IDE or Android Studio, Android SDK
If available Link to developer documentation/manual	<a href="https://github.com/suleimanyerima/MobiQ-survey-app/blob/master/MobiQApplicationMap-2.pdf">https://github.com/suleimanyerima/MobiQ-survey-app/blob/master/MobiQApplicationMap-2.pdf</a> <a href="https://github.com/suleimanyerima/MobiQ-survey-app/blob/master/installing-project-in-eclipse-luna.pdf">https://github.com/suleimanyerima/MobiQ-survey-app/blob/master/installing-project-in-eclipse-luna.pdf</a> <a href="https://github.com/suleimanyerima/MobiQ-survey-app/blob/master/APIs-online-guides-references.pdf">https://github.com/suleimanyerima/MobiQ-survey-app/blob/master/APIs-online-guides-references.pdf</a>
Support email for questions	<a href="mailto:syerima@dmu.ac.uk">syerima@dmu.ac.uk</a>

\* Corresponding author.

E-mail address: [syerima@dmu.ac.uk](mailto:syerima@dmu.ac.uk) (S.Y. Yerima).

## Software metadata

Current software version	v1.0
Permanent link to executables of this version	<a href="https://github.com/suleimanyerima/MobiQ-survey-app/blob/master/MobiQapp.apk">https://github.com/suleimanyerima/MobiQ-survey-app/blob/master/MobiQapp.apk</a>
Legal Software Licence	GPLv2
Computing platforms/Operating Systems	Android
Installation requirements & dependencies	
If available, link to user manual - if formally published include a reference to the publication in the reference list	<a href="https://github.com/suleimanyerima/MobiQ-survey-app/blob/master/App-configurations.docx">https://github.com/suleimanyerima/MobiQ-survey-app/blob/master/App-configurations.docx</a>
Support email for questions	<a href="mailto:syerima@dmu.ac.uk">syerima@dmu.ac.uk</a>

## 1. Introduction

MobiQ is a smartphone application developed to enhance longitudinal survey data capture. Traditional methods of data collection (i.e. paper-based surveys) can be costly to conduct over a long time period. In addition, as longitudinal surveys are completed by a cohort of participants at regular intervals such as annually, the long period between surveys may limit the extent to which dynamic phenomena such as social attitudes, behaviours, health status etc. are captured. In recent decades, one approach to this challenge has been the manufacture of bespoke devices, used for 'Ecological Momentary Assessment' (EMA, i.e., real-time administration of survey items) around phenomena which readily fluctuate [1–3]. While issuing devices such as accelerometers and pedometers to participants is standard in physical activity research where distance or acceleration are being measured [4,5], the quality of data is dependent on participant compliance in carrying the device. By comparison, smartphones are carried routinely, which means data collection via phone overcomes these problems [6]. In addition to enabling EMA on the android platform, MobiQ also offers a novel approach to measuring real-time social connection free from recall bias by using phone and text message contact as a measure of social interaction. MobiQ can also provide information about participants' geographical position and thus provides a measure of interaction with the built environment.

MobiQ incorporates a modular and flexible design with several re-useable core components, hence making it easily adaptable to suit different projects. The app has been built to transparently perform the following primary functions:

- (a) Administer questionnaires twice-weekly (Mondays and Thursdays) at regular intervals.
- (b) Collect anonymised destinations of calls and texts from the phone. Collect anonymised phone number, phone make and model.
- (c) Collect intermittent location reading via the phone's GPS.
- (d) Allow users to send comments to the research team. Enable users to send photographs to the research team.
- (e) Allow users to autonomously pause and/or resume app operation at any time.
- (f) Send all collected data to a secure online server when an internet connection is available.
- (g) In the absence of an internet connection, securely store the data until a connection becomes available.

MobiQ has been successfully applied in the field for a pilot study concerning youth social and behavioural research. The project (Using smartphones to enhance longitudinal survey methods) was undertaken to explore the novel application of smartphones in enhancing longitudinal surveys in the social sciences. Since the app was developed primarily to collect data over a long period of time, the design has also been leveraged in other smartphone-based data-driven projects outside of the social sciences, for example in [7,8] and [9] for behaviour-based user implicit authentication.

## 2. Motivation and significance

Smartphones potentially provide an effective tool for objective measurements of social interactions in ways that could offer novel opportunities beyond the capabilities of traditional paper survey methods. As Eagle noted, the devices which people carry for communication purposes can effectively serve as sensors, providing insight into relational dynamics of individuals [10]. Traditional data collection tends to provide a 'snapshot' of the social world, rather than capturing the 'moving parts'. Social research relies on administering surveys at discrete intervals giving little or no insight into incremental change in social ties and individual behaviour between intervals. Depiction of the social world from research data are typically devoid of personal, social or geographical context, and liable to recall or information bias [11]. With the use of smartphones, a dynamic, rich and multi-faceted dataset can be obtained by collecting geographically sensitive social network data, regularly elicited self-reports of behaviour and participant-contributed photo-journals. Furthermore, the use of smart-phones enables cost-effective and time-saving methods to observe social networks over a sustained period.

Motivated by the need for a multi-faceted dataset, time-saving potential and cost-effectiveness, our interdisciplinary research project investigates using smartphones to enhance longitudinal survey methods. The project draws on expertise from Social Sciences, Geography, Computing, Digital Humanities and Public Health with the aim of assessing whether smartphone apps could benefit research projects across human sciences. The primary research question the project sought to answer was: to what extent can data collected using a smartphone improve our understanding of social worlds, simultaneously measuring (i) social networks and their operation, (ii) individual behaviour and (iii) how these relate to the physical environment. A secondary research question asked was: what are the ethical considerations to performing these studies, particularly in relation to the pilot test case of collecting information about illicit drug use. While other online surveys and survey phone applications were available (e.g. Qualtrics [12], Bristol Online Surveys [13], ODK (Open Data Kit) [14]), none of these met all the project requirements for: periodic and continuous survey data elicitation, passive GPS and social contact data monitoring, and the security of data collection and handling. For these reasons, the team built MobiQ, a bespoke Android application for the project. The design concepts are transferrable to other mobile OS platforms. Additionally, since MobiQ is modular, its passive elements could be integrated with other survey applications like ODK which is also open source.

The team is releasing MobiQ as an open source tool to aid other researchers who might be interested in exploring research projects from any discipline that could benefit from smartphone based data collection. MobiQ has been successfully applied for the study presented in [15]. Other recent findings made in our research studies involving MobiQ collected data will be reported in separate publications. In this paper, our main focus is on the design and implementation of the MobiQ software.

### 3. Software description

MobiQ implements a number of functionalities which allows it to continuously run in the background (for the duration of the survey) and periodically collect and upload data to a secure online location (i.e. a secure database created specifically to store the collected data). This innovative design ensures that the app is battery efficient by calling on functionalities to operate only when needed.

Additionally, despite being equipped with real-time data upload capability, MobiQ has the ability to operate without network connectivity, so that users have the option to enable data uploads only when connected to a Wi-Fi network for example.

#### 3.1. MobiQ background components and app states

MobiQ is written in Java and was built on the Eclipse Android SDK platform. MobiQ implements at its core, a Broadcast receiver class called *MobiQBroadcastReceiver*. This acts as a watchdog that coordinates other key functionalities within the application. *MobiQBroadcastReceiver* listens for intents (which signal events) from an implemented *AlarmManager* and also a `BOOT_COMPLETED` intent from the system. The `BOOT_COMPLETED` intent signals a device reboot, and by using this to trigger the app operation each time it is received, continuous app operation is ensured as long as the app is kept in the activated state (i.e. not disabled/suspended by the user).

When the app is installed for the first time, *MobiQBroadcastReceiver* starts the *AlarmManager*. Whenever the device is restarted, *MobiQBroadcastReceiver* also restarts the *AlarmManager*, except when the MobiQ app has been explicitly disabled/suspended by the user (i.e. the app is in the suspended state of operation). The user can re-activate the app operation at any time and this also explicitly restarts the *AlarmManager*. The *AlarmManager* is implemented to send a signal to the *MobiQBroadcastReceiver* periodically (every 10 min). Since this signal is an intent, it incurs very low overhead thus enabling continuous app operation with a very high degree of power efficiency. Thus, instead of continuously running as a service in the background the app is effectively 'asleep' until an *AlarmManager* signal is received. When *MobiQBroadcastReceiver* receives a signal from the *AlarmManager*, the type of event that is triggered depends on the time elapsed since it was last invoked. Therefore, *MobiQBroadcastReceiver* maintains the state information of each event by storing the previous times the events were triggered. The events that are scanned for include: GPS location information (every 20 min), outgoing calls updates (every 60 min), outgoing SMS updates (every 120 min), repeated surveys (every Monday and Thursday afternoons). Note that the *AlarmManager*'s periodic signal is set to 10 min in order to capture the GPS data more accurately (since a 20 min interval for GPS data capture was set by the requirements of our pilot studies). If GPS data is required less frequently, say every hour, then the *AlarmManager* can be reconfigured to send its intent signals every 30 mins thus allowing the app to 'sleep' for longer intervals.

Fig. 1 shows a state diagram of the MobiQ app operation. The app can be in the activated, suspended, or device off states. When the app is suspended, no background data collection occurs and the survey questionnaires will also not be presented to the user. The device off state is registered by the app in order to allow *MobiQBroadcastReceiver* to continue the app operation in the last known state.

Normal app operation occurs during the 'activated' state. In this state, *MobiQBroadcastReceiver* listens for event signals from the *AlarmManager* as depicted in the flowchart in Fig. 2. The periodically generated event signal allows different events to be recognised and various corresponding 'Managers' invoked to collect,

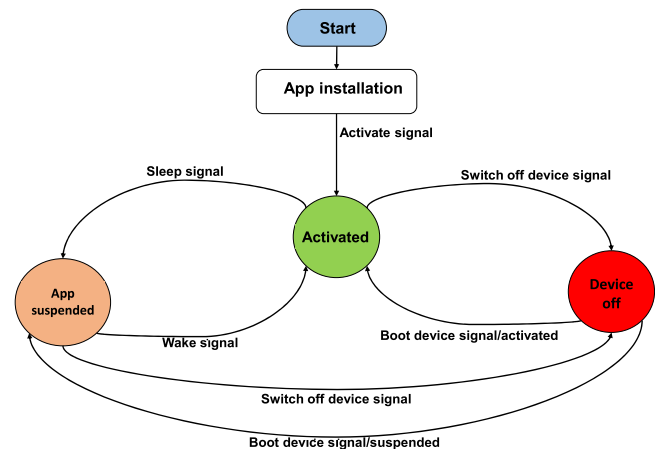


Fig. 1. MobiQ operational state diagram.

update, store, and/or upload different types of data to the remote database on a secure server.

These managers are implemented as background services using the Android services class. The implementation of the managers as service components is mandated by the need for remote connectivity and communications where the HTTP protocol is used to post data to the remote database. These managers are implemented as the following classes: *LocationHttpService*, *CallLogHttpService*, *SmsLogHttpService*, *MondayQHttService*, and *ThursdayQHttService* respectively. Each of these is invoked according to the conditions illustrated in Fig. 2. A similar service called *DeviceInfoHttpService* is also implemented to post device information to the remote database, but this is only invoked once during the app installation in order to register the device as part of the survey so as to be able to correctly link all collected data to their originating devices.

All of these services employ *ConnectionManager*, *FileUploader* and *FilesChecker* objects. These are code elements (classes) designed to be shared/utilised by any background service implemented within the MobiQ app. *ConnectionManager* is designed to use the Android *ConnectivityManager* API to get network information and connectivity status. This makes it possible to recognise whether the device is connected to a WiFi network or a mobile network and to also obtain the status of this connection. The *FilesChecker* implements methods to check data file contents. It uses the information received from the *ConnectionManager* object to decide whether data can be uploaded. If there is valid data in any of the files, it employs a *FileUploader* object to upload the file to a given remote URL (i.e. the remote database location).

#### 3.2. MobiQ user interactive components

##### 3.2.1. Main activity screen

Once the MobiQ app has been installed, it goes into activated mode with all the background events occurring continuously. The initial survey questionnaires are brought up to be answered by the users at the start of the survey immediately after installation. When that is completed, the user then exits the app and subsequent questionnaires will pop up as scheduled (i.e. Mondays and Thursdays).

The user can also interact with the app at any time by launching the main activity screen using the app icon. The top left screen shot in Fig. 3 shows the user interface activity that comes up when the app is launched by the user. This activity has a button that allows the user to navigate to a screen where they can report any problem with the app operation. This allows any problems encountered with the app to be notified and remedial actions to be

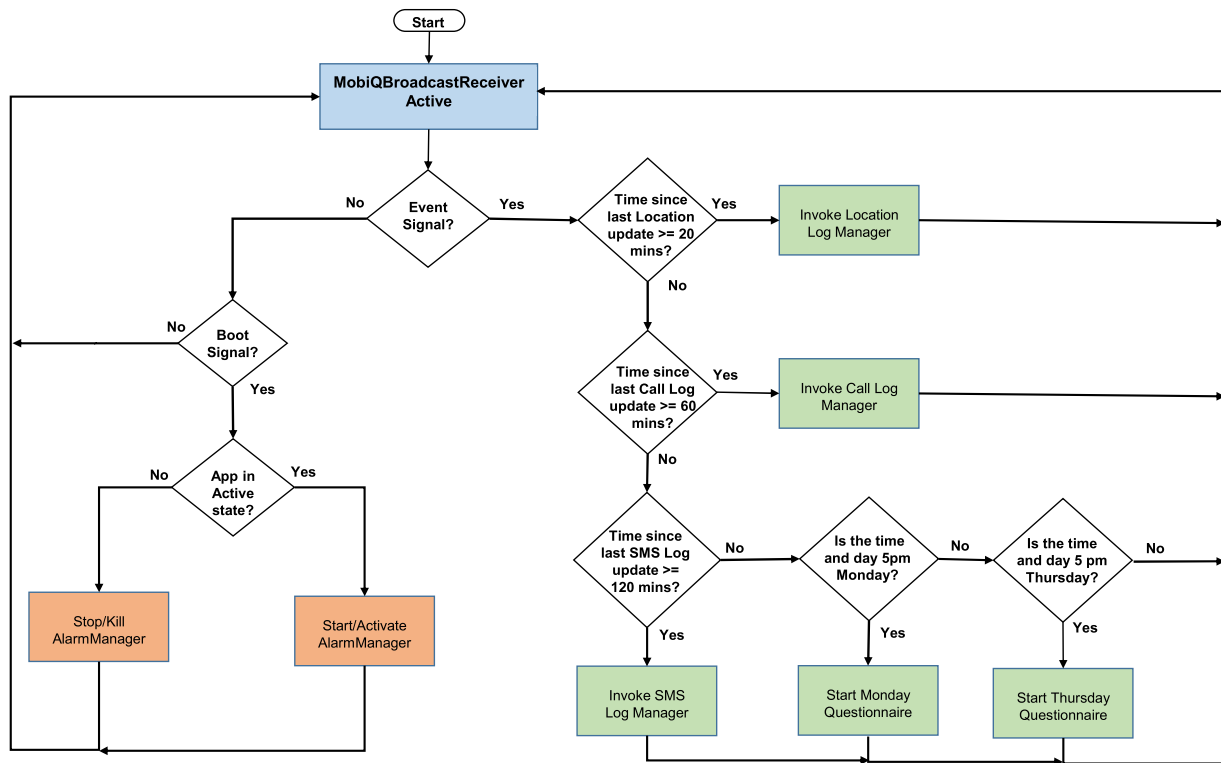


Fig. 2. Flowchart of the MobiQBroadcastReceiver operation.

taken where possible by the survey administrators. There is also a button for pausing/suspending the app operation, indefinitely. The same button toggles to allow the user to resume operation again when desired. The third button allows the user to navigate to the screen where a picture can be taken and uploaded to the survey database. The fourth button on the main screen allows for the selection of an Administrator operation mode. It brings up the top right screen shot shown in Fig. 3. This operation mode requires a password to login as shown in the bottom left screen shot. The Admin mode enables the survey administrators to operate the app in a 'privileged mode' that can be used to demonstrate the features and operation of the app to potential survey participants. It enables the app to be operated in a 'demo' mode which is inaccessible to the regular users of the app during field operation.

### 3.2.2. Survey questionnaires

The questionnaires are also implemented as Android Activities. Each survey questionnaire is invoked at a designated time as shown in the flowchart of Fig. 2. However, it is preceded by a dialogue box that can allow the user have the option to take the survey immediately or to postpone it by pressing a 'remind me later' button as shown in Fig. 4. When a survey is completed, either the *MondayQHttpService* or the *ThursdayQHttpService* is invoked to upload the data immediately or store it locally as explained in the previous section.

In order to allow for easy modification and reusability, most of the survey questions are not hard-coded onto the activities that display the questionnaire screen. Instead, they are loaded on demand from the *sstrings.xml* resource file located in the app's *resources* folder. The questions are grouped into string arrays according to their categories. Fig. 5 illustrates this with code example from one of the MobiQ questionnaire activities. It shows how the `getResources().getStringArray()` object is used to dynamically load questions from a `non_club_array` string array containing survey questions related to non club activities. In the *for loop*, the

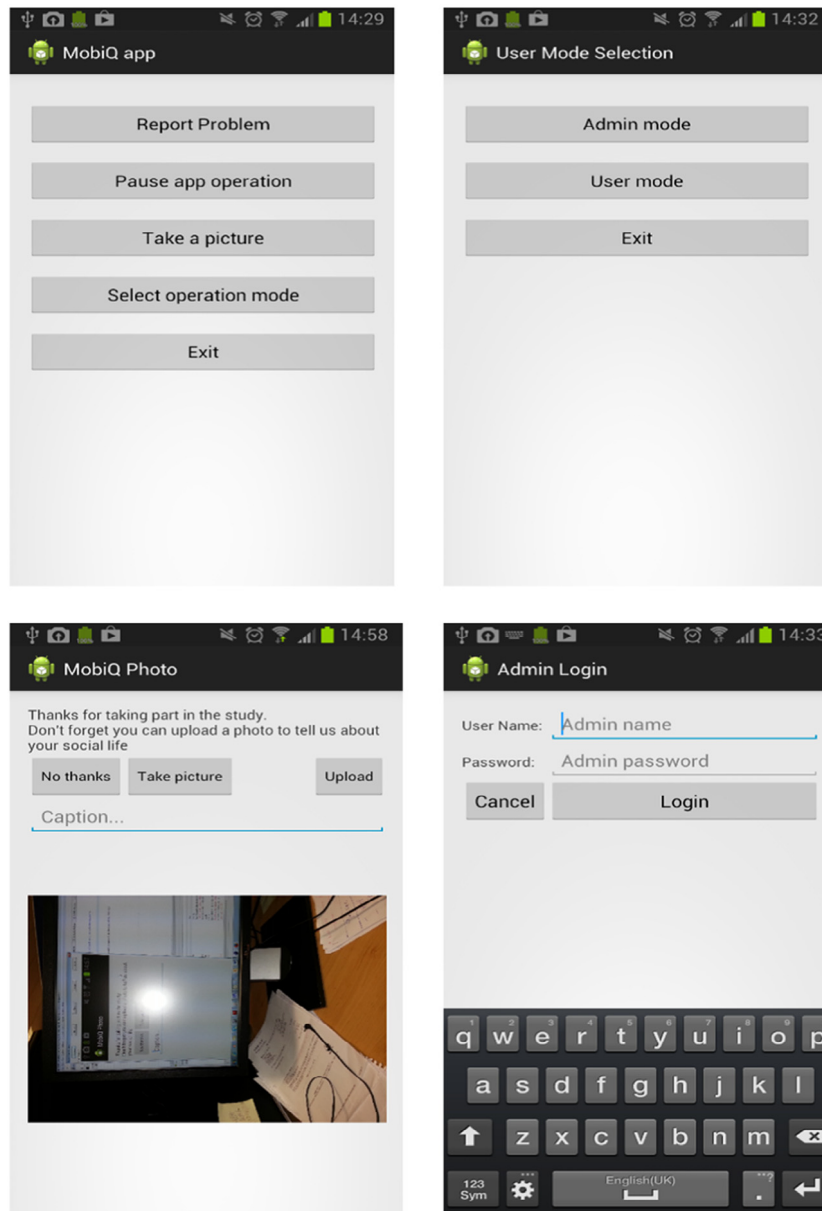
questions are subsequently displayed as text next to checkboxes. In addition to the questions, the URL addresses of the server files that receive the uploaded data for processing and storage in the remote database are also listed in the *sstrings.xml* resource file. This implementation allows the remote data upload location as well as the survey questions to be easily modified to accommodate changes and allow code re-use with minimal modification.

### 3.2.3. Remote server components

MobiQ is designed to continuously upload data to a remote database located on a secure server. In order to accomplish this, two key components were used in the back-end; Web server scripts and a structured database. MobiQ can be used with any Web server scripting language and any database format. In our project we used PHP for the server side scripts and MySQL for the database. Both of these were installed on an Apache web server running on a Windows 2012 R2 cloud server. The database (*mobiq*) incorporates the following tables: *deviceinfo*, *location*, *callslog*, *sm-slog*, *mondaysurvey*, *thursdaysurvey*, *reportproblem*, and *photosinfo* respectively. Each of the tables store different data (if available) at intervals corresponding to the flowchart depicted in Fig. 2. The fields created for each table are shown in Table 1, while Fig. 6 shows the *thursdaysurvey* table with uploaded sample data from survey users.

As can be seen from Table 1, the device imei is essentially used as the primary key that connects all of the tables in a relational database structure. The id field in the device info table also stores the device imei. Note that all of the personally identifiable information (i.e. imei, phone numbers) are encrypted before storing and/or uploading the data.<sup>1</sup> Also, data file storage on the phone is private, so that the user cannot see or access the collected data.

<sup>1</sup> The encryption key can be changed in the cipher module of the MobiQ source code.



**Fig. 3.** User interactive components of the MobiQ app.

**Table 1**  
MobiQ database tables and their corresponding fields.

Table name	Fields
Device info	id, number, osname, osver, osrelease, model, cpu, api, time
Location	imei, latitude, longitude, time
Callslog	imei, deviceNumber, destinationNumber, time, duration
Smslog	lmei, deviceNumber, destinationNumber, time
Mondaysurvey	imei, time, Q1, Q2, Q3 ...
Thursdaysurvey	imei, time, Q1, Q2, Q3 ...
Reportproblem	imei, time, problem
Photosinfo	imei, time, latitude, longitude, photolink, comments

#### 4. Impact and conclusions

As far as we know, there are no existing open source apps or frameworks accessible to social sciences researchers that offer the range of functionalities of the MobiQ app. MobiQ provides an easy to use tool for conducting social research on a large scale, while collecting data longitudinally with minimal fieldwork resources. The ability to collect data without information being stored on

the phone itself makes it particularly suited to collecting data where concerns exist over data security or the possibility of others accessing information on the phone.

In our project, MobiQ has been applied in the field where 200 secondary school pupils (16–18 year olds) from two schools in Northern Ireland were invited to take part in a pilot study (following ethical approval granted by the School ethics committee). The study was focused on substance use and social activities [15].

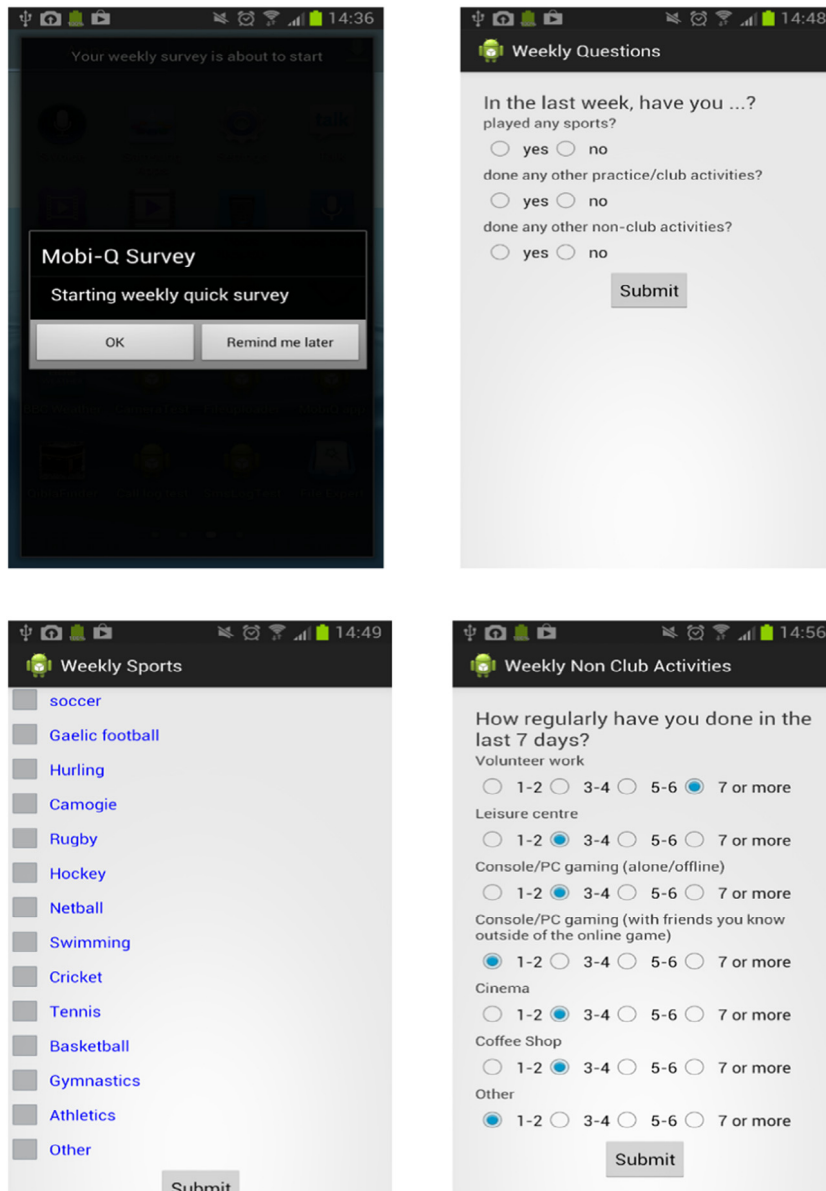


Fig. 4. Sample questionnaires from the MobiQ app.

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.List_Layout);
    checkboxLayout = (LinearLayout) findViewById(R.id.Checkbox_Layout);
    data = getResources().getStringArray(R.array.non_club_array);

    for (int i = 0; i < data.length; i++) {
        CheckBox cb = new CheckBox(getApplicationContext());
        cb.setText(data[i]);
        cb.setTextColor(Color.BLUE);
        checkboxLayout.addView(cb);
    }
}

```

Fig. 5. MobiQ example activity questionnaire code snippet illustrating dynamic loading of questions from *sstrings.xml* resource file.

The app was pre-programmed to administer questions on Monday afternoons (substance use) and Thursday afternoons (leisure activities), with calls and texts messages logged as indicators of social interaction. The app successfully collected and stored anonymised data from participants over a period of six months. The outcomes

of this pilot study demonstrated a successful achievement of the proof-of-concept.

The data collected from the app has also been applied to a study investigating GIS spatial analysis of the relationship between spatial movements and behavioural patterns [16]. This study concluded that the data provided a novel method to analyse patterns of movement in relation to social demographics, social settings and lifestyle choices.

Although MobiQ was developed to provide longitudinal survey capabilities on smartphones, with enriched contextual data capture to enhance social sciences research, the software project has had an impact on research in other areas as well. The successful field application of the software on survey participants' smartphones inspired the re-use of the architecture in other smartphone-based projects. For example, the architecture has been used in the design of a data collection app for a data-driven smartphone-based implicit authentication project overseen by a member of the MobiQ development team. In particular, the MobiQ Broadcast receiver-Alarm manager architecture provided the basis for the app implemented to investigate *event-driven implicit*

sn	imei	time	anySport	anyClubs	anyNonClubs	Soccer	GaelicFootball	Hurling	Camogie	Rugby	Hockey	Netball	Swimming	Cricket	Tennis	Basketball	TableTennis	Volleyball
1		2014-03-06 17:17:17	yes	yes	yes	none	1-2	none	none	none	none	1-2	none	none	none	none	none	none
2		2014-03-06 17:23:54	yes	no	yes	3-4	none	none	none	none	none	none	none	none	none	none	none	none
3		2014-03-06 17:14:08	yes	yes	yes	none	1-2	none	none	none	none	none	none	none	none	none	none	none
4		2014-03-13 17:08:07	yes	no	yes	1-2	none	none	none	none	none	none	none	none	none	none	none	none
5		2014-03-13 17:21:11	no	no	yes	none	none	none	none	none	none	none	none	none	none	none	none	none
6		2014-03-13 17:27:35	yes	yes	yes	none	1-2	none	none	none	none	none	none	none	none	none	none	none
7		2014-03-20 17:02:06	yes	yes	yes	none	1-2	none	none	none	none	1-2	none	none	none	none	none	none
8		2014-03-20 17:04:41	no	no	no	none	none	none	none	none	none	none	none	none	none	none	none	none
9		2014-03-20 17:14:53	no	yes	yes	none	none	none	none	none	none	none	none	none	none	none	none	none

Fig. 6. Sample weekly survey stored in the thursdaysurvey table of the MobIQ database.

authentication schemes for mobile access control as reported in [7,8], and [9]. This demonstrates that the MobIQ software can have potential impact beyond application in the social sciences. We are therefore motivated to make the MobIQ source code public so that researchers across multiple domains may benefit from our contribution.

#### 4.1. Other ways to reuse the MobIQ software

As mentioned earlier, MobIQ has several re-useable components, including its core design architecture. Other possible ways the app could be adapted include: (a) leveraging the existing services such as *CallLogHttpService* to log and upload other types of data. (b) Modifying the activities for a different set of questions. (c) Employing a different backend architecture or a different web application/database technology while leaving the app unchanged (except for changing the URL addresses in the *strings.xml* file). (d) Using a *webview* to link to online questionnaires instead of administering the question on-phone using activities. Note that this option will not allow for offline questionnaires but could suffice for some projects. (e) Combining user supplied, sensor or communications data with social media data obtained from other social media survey apps [17]. (f) Complementing data obtainable from open source toolkits such as ODK [14] with passive sensor or communications data.

#### Acknowledgements

MobiQ was developed as part of the project: Using smartphones to enhance longitudinal survey methods, funded by an ESRC Transforming Social Sciences Grant (ES/L003198/1). Mark McCann holds a Medical Research Council/University fellowship supported by MRC partnership grant (MC/PC/13 027), and Medical Research Council and Chief Scientist Office through the SPSU Complexity in Health Improvement (MC\_UU\_12017/14/SPSU14) and Social Relationships and Health Improvement (MC\_UU\_12017/11/SPSU11) programmes. Helen McAneney wishes to acknowledge the Medical Research Council

Research Methodology Fellowship (G0902112), which funded her during this project.

#### References

- [1] Shiffmann S, Stone AA, Hufford MR. Ecological momentary assessment. *Annu Rev Clin Psychol* 2008;4:1–32.
- [2] Yoshiuchi K, Yamamoto Y, Akabayashi A. Application of ecological momentary assessment in stress-related diseases. *Biopsycosoc Med* 2008;2:13.
- [3] Moskowitz DS, Young SN. Ecological momentary assessment: what it is and why it is a method of the future in clinical psychopharmacology. *J Psychiatry Neurosci* 2006;31(1):13–20.
- [4] Tudor-Locke C, et al. How many steps/day are enough? for children and adolescents. *Int J Behav Phys Act* 2011;8:78.
- [5] Almanza A, et al. A study of community design, greenness and physical activity in children using satellite, GPS and Accelerometer Data. *Health Place* 2012;18(1):46–54.
- [6] Runyan JD, et al. A smartphone ecological momentary assessment/intervention “app” for collecting real-time data and promoting self-awareness. *PLoS One* 2013;8(8):e71325.
- [7] Yao F, Yerima SY, Sezer S. Event-driven implicit authentication for mobile access control. In: Proc. next generation mobile applications, services and technologies; 2015.
- [8] Yao F, Yerima SY, Kang B, Sezer S. Continuous implicit authentication for mobile devices based on adaptive neuro-fuzzy inference system. In: Proc. 2017 international conference on cyber security and protection of digital services; 2017.
- [9] Yao F, Yerima SY, Kang B, Sezer S. Fuzzy logic based implicit authentication for mobile access control. In: Proc. SAI computing conference; 2016.
- [10] Eagle N, Petland A, Lazer D. Inferring friendship network structure using mobile phone data. In: PNAS. USA; 2009.
- [11] Schulz KF, Grimes DA. Case-control studies: research in reverse. *Lancet* 2002;359(9304):431–4. PMID 11844534, [http://dx.doi.org/10.1016/S0140-6736\(02\)07605-5](http://dx.doi.org/10.1016/S0140-6736(02)07605-5).
- [12] Qualtrics: Available from: <https://www.qualtrics.com/>. [Last Accessed April 2018].
- [13] Online Surveys: Available from: <https://www.onlinesurveys.ac.uk/>. [Last Accessed April 2018].
- [14] Open Data Kit: Available from: <https://opendatakit.org>. [Last Accessed April 2018].
- [15] McAneney H et al. Smartphones- a method for longitudinal survey data capture: One step forward or two steps back?. In: Complex systems, health disparities and population health: Building bridges.
- [16] Barry L. GIS for longitudinal survey evaluation, School of Geography, Archaeology and Paleocology, Queen’s University Belfast, Unpublished report; 2014.
- [17] Bail CA. Taming big data: Using app technology to study organizational behavior on social media sites. *Sociol Methods Res* 2015;1–29.