Towards a Methodology and a Toolkit to Analyse Data for Novices in Computer Programming

Tatiana Person, Iván Ruiz-Rube, and Juan M. Dodero

Escuela Superior de Ingeniería (Puerto Real, Cádiz), University of Cádiz, Spain
{tatiana.person,ivan.ruiz,juanma.dodero}@uca.es

Abstract. The incorporation of mobile applications in diverse environments generates a large amount of information resulting from the interaction of users with these mobile applications. The analysis of this information can facilitate decision-making or evaluation of the process for the professionals, allowing for improved results or the detection of certain patterns. There are multiple technologies for data analysis that can be applied to analyse the captured information. However, the development of mobile applications that incorporate these features is not trivial for a user who does not have the appropriate programming skills. In addition, decision-making to select the data analysis technology to be applied in each situation is difficult for this type of users. In this work, a methodology is presented to help people without appropriate programming skills in the above process. Finally, this methodology is being applied in a visual authoring tool to enable the users to create mobile applications that incorporate this features.

Keywords: Authoring tools · Data analysis · Learning analytics · Mobile learning

1 Introduction

According to the 2017 Ditrendia report [1], nowadays 66% of the world's population has a smart-phone and the use of mobile applications represents 60% of the time spent in the digital world. Based on the above data, we can conclude that mobile applications are playing an increasingly important role in people's daily lives. Existing digital content repositories, such as *Google Play Store* or *App Store*, contain mobile applications in a variety of topics: applications for people-to-people communication, to entertainment, to life-style control and monitoring. Mobile applications also emerged for different educational purposes: explaining specific topics or concepts, evaluating students, conducting laboratory experiments, solving exercises collaboratively, learning foreign languages, etc.

New ways of interaction between users and mobile devices such as the use of elements for verbal interaction through the use of voice commands and soundtracks [9], touch through the use of tactile surfaces and haptic devices [11], or gesture by capturing human movement [17] are generating a large amount of

data [13]. This data may be collected and later processed using *Learning Analytics* (LA) and *Educational Data Mining* (EDM) techniques. Which allow us to evaluate the experience and learning of users, as well as the usability of the applications themselves [3],[4], because the data can be transformed into information and knowledge. Thus, several specifications related to learning analytics have been developed, such as *Learning Tools Interoperability* (LTI), to facilitate the integration of e-learning tools; *Experience API* (xAPI), for the publication of meta-data on real learning activities; or the most recent *Caliper Analytics*, for data extraction and computation of metrics [8].

In addition, these data can be analysed using the techniques covered by the *Big Data Analytics* concept. The choice for one of these techniques will depend on the characteristics contemplated in the context in which the data are analysed, including the time that they were created [6]. First, if the analysis of the data generated in the past and stored in databases at the time of analysis is required, relational database analysis, non-relational database analysis or OLAP analysis may be used. Second, if the analysis of data generated during the analysis is needed, stream analytics or complex event processing may be applicable. Finally, if the analysis of data generated in the past and stored in the database at the time of the analysis is demanded, but this analysis intends to predict the behaviour or some future data characteristic, machine learning or deep learning may be applicable.

However, it is important to be aware of the difficulty of conducting large amounts of data analysis for users without extensive programming knowledge. In this sense, the use of environments that support such users to perform these types of operations is essential. Therefore, the main aim of this research is to design a methodology that allows non-technical users to make use of this type of analysis of large amounts of data independently, without requiring IT experts to do this work for them. Based on the results described in [7], the author concludes that possibly one of the main challenges of the data processing lifecycle is having the ability to choose which tools and technologies to use effectively and efficiently. In addition, the necessary characteristics for the construction of this type of architecture must also be considered [15].

Finally, the rest of the work is structured as follows: in the second section the background is presented. The third section presents the initial version of the designed methodology. The fourth section introduces the implementation of the VEDILS components to provide the functionalities proposed by the methodology. Finally, the conclusions of this work are presented in section five.

2 Background

Firstly, visual authoring tools are computer applications that facilitate the creation, publication and management of multiple materials in digital format. Normally, these tools model software development processes that require IT experts to be able to execute them, allowing non-expert users to do this work with the support of the automation provided by the tool. For example, with *Google*

Forms[1], non-web development experts can easily create web surveys that they can share and then analyse the data they collect. There are application authoring tools that use a visual language to create these applications, such as *Scratch*[2], *MIT App Inventor*[3], *Pocket Code*[4] or *VEDILS*[5]. On the other hand, there are application authoring tools that use a textual language, such as *Microsoft Touch Develop*[6], *Upplication*[7], *GameSalad*[8] or *Alice*[9].

Secondly, context can be defined as a fragment of information that can be used to characterise the situation of a participant in an interaction[2]. In addition, by detecting context information, applications can present useful information related to the context of users and adapt their behaviour to changes in the environment[14]. In the [16] work, the following are defined as context properties: domain context, location context, data context and user context.

Finally, the technologies that can be used for data analysis can be classified according to when the data to be analysed is created, as follows:

- **Technologies for Past Data Analysis**: technologies to analyse previously created and stored data.
  - *Relational database analysis*: First, relational DBMS (Database Management System) are storage systems that comply with the relational model. In this type of system, the databases are composed of several tables and relationships with unique names. The relationships between tables are made through the use of primary and foreign keys. On the other hand, the SQL language is used to manipulate and consult the information stored in a relational DBMS. Some of the most used relational DBMS are: *MySQL*[10], *Oracle*[11] or *PostgreSQL*[12].

  - *Non-Relational database analysis*: On the other hand, non-relational DBMS require less powerful machines, facilitating horizontal scalability; improving system performance requires only the addition of new nodes. They can store large amounts of data, through their distributed structures and do not generate bottlenecks. On the other hand, the most popular non-relational DBMS today include the following: *Redis*[13], *Cas-*

[1] https://www.google.es/intl/es/forms/about/
[2] https://scratch.mit.edu
[3] http://appinventor.mit.edu
[4] https://share.catrob.at/pocketcode/
[5] http://vedils.uca.es
[6] https://www.touchdevelop.com
[7] https://www.upplication.com
[8] http://gamesalad.com
[9] https://www.alice.org
[10] https://www.mysql.com
[11] https://www.oracle.com
[12] https://www.postgresql.org
[13] https://redis.io

sandra[14], *CouchBase*[15], *HBase*[16], *MongoDB*[17] and *AllegroGraph*[18].

- *Multidimensional database analysis*: OLAP (On-Line Analytical Processing) is considered as a solution used in Business Intelligence that allows the user to extract and visualise data from a variety of points of view. To perform this type of analysis, multidimensional structures are used, represented metaphorically as a cube (OLAP cube) whose cells correspond to events that occur in the business domain. Currently, existing tools that incorporate the ability to perform OLAP analysis include: *Pentaho BI*[19], *Dundas BI*[20], *Sisense*[21], *Domo*[22] and *Tellius*[23].

– **Technologies for Present Data Analysis**: technologies to analyse data that are created during the course of the analysis.

- *Stream analytics*: With reference to Stream analytics technology, the tools for data transmission and the tools for data processing in streaming should be considered. First, as tools for the transmission of data, the Message Queuing or Pub-Sub Messaging systems must be considered, which are used as asynchronous intermediary systems, implementing the publishing-subscribing paradigm. Typically, these systems decouple different components of an architecture. Popular message queuing systems include: *Apache Kafka*[24], *RabbitMQ*[25] and *ActiveMQ*[26]. On the other hand, streaming data processing tools work with continuous and non-persistent data streams. This data can come from sensors or publications on social networks, for example. Currently, streaming data processing tools include: *Apache Flink*[27], *Apache Spark*[28] and *Apache Storm*[29].

- *Complex Event Processing*: Complex event processing (CEP) systems differ from streaming processing systems in that they associate semantics with the information they are processing, which are notifications of events occurring in the external world and observed in information

[14] http://cassandra.apache.org
[15] https://www.couchbase.com
[16] https://hbase.apache.org
[17] http://www.mongodb.com
[18] https://allegrograph.com
[19] http://www.pentaho.com
[20] http://www.dundas.com/dundas-bi
[21] https://www.sisense.com
[22] https://www.domo.com
[23] http://www.tellius.com
[24] https://kafka.apache.org
[25] https://www.rabbitmq.com
[26] http://activemq.apache.org
[27] https://flink.apache.org
[28] https://spark.apache.org
[29] http://storm.apache.org

sources. The CEP engine is responsible for filtering and combining such notifications to understand what is happening in terms of complex events (high-level events composed of the combination of simple events) and then notifying customers subscribed to these alarms[10]. Currently, complex event processing tools include *Apache Spark*, *Apache Flink* and *Esper*[30].

– **Technologies for Future Data Analysis**: technologies to analyse data to be created later.

  • *Machine Learning*: can be defined as a mechanism for searching for patterns and creating intelligence on a machine enabling you to learn. This means that you will be able to solve problems more correctly or efficiently in the future based on your experience, just as in humans. Machine Learning algorithms can be classified according to the type of learning problem they solve in: classification, grouping, regression, optimisation and simulation. On the other hand, the analyses applied using Machine Learning can be classified according to the behaviour of the algorithm in: Supervised learning, Unsupervised learning, Semi-supervised learning or Reinforcement Learning[12]. Currently, tools for applying Machine Learning algorithms include: *Apache Mahout*[31], *R*[32], *Julia*[33], *Apache Spark* and *Apache Flink*.

3 Methodology

This work proposes the first version of a methodology to assist in decision-making when choosing a technology for data analysis. Learning analytics is an emerging area within e-learning and generally consists of the following stages: capture, report, predict, act and refine[5]. Following the above, this methodology should allow programming novices to perform these steps without the assistance of computer experts. In addition, support should also be provided so that they can easily select what type of technology to apply to the reporting of the data they have captured. Figure 1 shows a diagram with possible decisions. The following context characteristics have been used to decide on the use of each of the technologies:

– *Temporal instant*: Temporal instant when the analysis is performed.
– *Data structure*: Expected characteristics in the recorded data structure.
– *Query format*: Expected characteristics of the queries to be made.

[30] http://www.espertech.com
[31] http://mahout.apache.org
[32] https://www.r-project.org
[33] https://juliacomputing.com

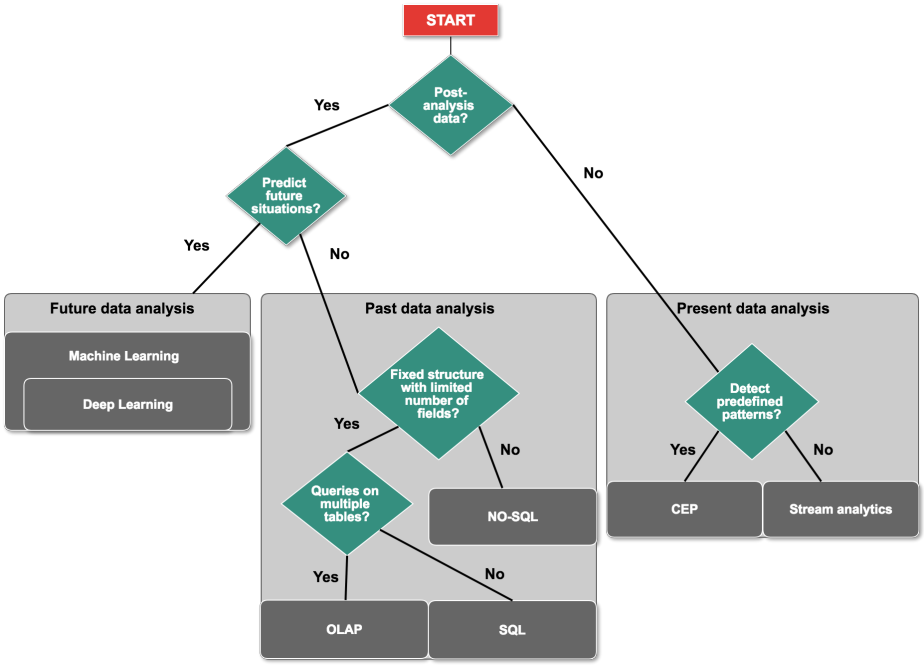


Fig. 1: Methodology to assist in decision-making by choosing a data analysis technology.

4 Developed components

To enable the data analysis to people without programming knowledge, the proposed methodology has been applied in a visual application authoring tool. VEDILS is an environment based on MIT App Inventor 2 to easily develop multimodal and interactive learning scenarios. The platform includes a view where the user can design the user interface and a view (Blockly-based editor) where they can define the behaviour of the elements included in the applications. VEDILS provides a set of additional features that can be integrated with those already provided by MIT App Inventor 2. Features such as augmented reality, virtual reality, gestural interaction and learning analytics, among others are available. The set of components implemented for this research are described below. Currently, the implemented components only support the past and present data analysis functionalities of the methodology.

4.1 Enabling the recording of information to analyse

First, we have developed the ActivityTracker component, which aims to facilitate the process of recording information related to interactions between users and mobile devices and finally register it in a database. Currently, the stores

supported by this component are Google Fusion Tables (SQL) and MongoDB (NO-SQL). In addition, this component allows the streaming processing of the sent data. The above options can be configured from the ActivityTracker component properties (see Figure 2).

On the other hand, ActivityTracker allows automatic registration every time a function is invoked, a property is accessed or an event is triggered from any of the existing VEDILS components while using the application (see Figure 3a). In addition, it allows you to send data with a semantics previously defined by the application designer (see Figure 3b).

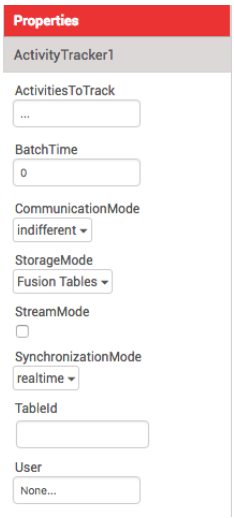
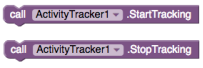
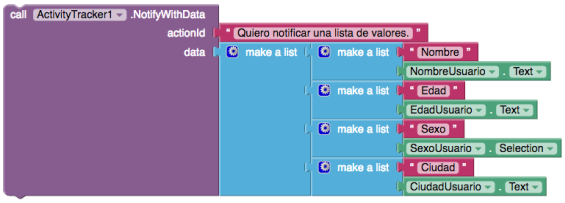


Fig. 2: Properties to configure the ActivityTracker component



(a) Blocks to record automatic notifications

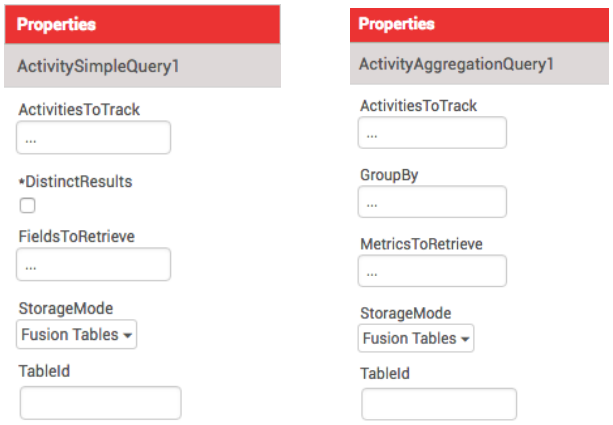


(b) Blocks to record user-specific notifications

Fig. 3: Blocks to record notifications using ActivityTracker component

4.2 Enabling to query the recorded information

Secondly, the objective of the components *ActivitySimpleQuery* and *ActivityAggregationQuery* is to allow the consultation of the data recorded in the database by the component *ActivityTracker*. The *ActivitySimpleQuery* component allows for simple queries, i.e. queries that include data selection and filtering (*select*). On the other hand, the *ActivityAggregationQuery* component allows you to perform more complex queries, in which you can group data (*groupBy*) and perform metrics on them, such as: arithmetic mean, sum, maximum, minimum or number of elements. The above options can be configured from the ActivitySimpleQuery and ActivityAggregationQuery components properties (see Figures 4a and 4b). The queries made can be conventional queries (see Figure 5a) that are executed only once or streaming queries (see Figure 5b) that are executed iteratively in time intervals.



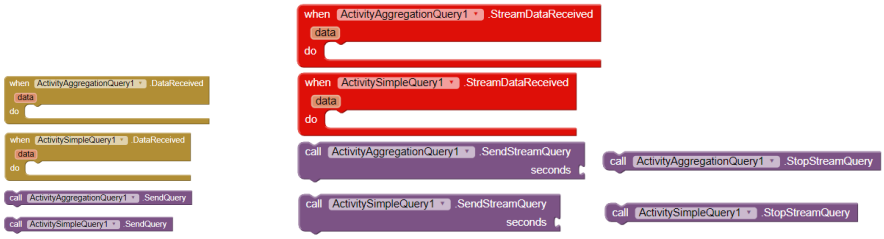
(a) Properties of ActivitySimpleQuery component

(b) Properties of ActivityAggregationQuery component

Fig. 4: Properties to configure the ActivitySimpleQuery and ActivityAggregationQuery components

4.3 Enabling the representation of obtained-query results

The purpose of the *Chart* and *DataTable* components is to present a specific data set in graphical or tabular form, which will be the result of a previous query using the *ActivitySimpleQuery* and *ActivityAggregationQuery* components. The *Chart* component enables you to represent the data in graphic format (row, column, etc.). The above options can be configured from the DataTable and

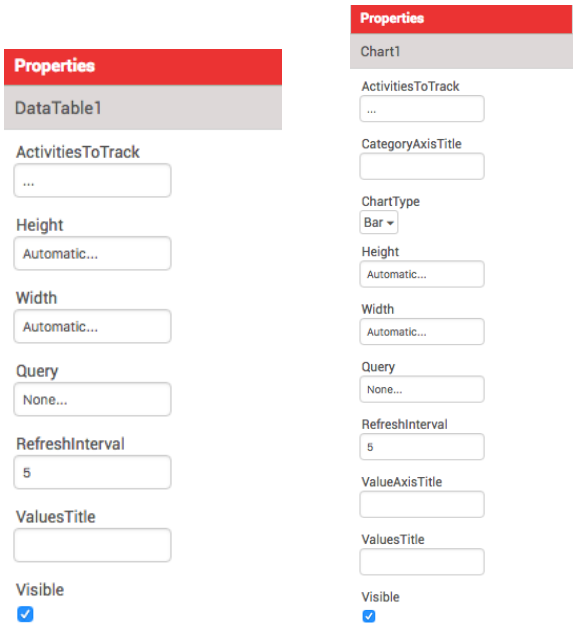


(a) Blocks to process conventional queries

(b) Blocks to process streaming queries

Fig. 5: Blocks to process queries using ActivitySimpleQuery and ActivityAggregationQuery components

Chart components properties (see Figures 6a and 6b). And on the other hand, the *DataTable* component allows you to represent the data in table format (see Figure 7).



(a) Properties of DataTable component

(b) Properties of Chart component

Fig. 6: Properties to configure the DataTable and Chart components

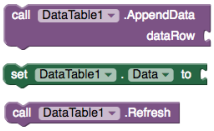


Fig. 7: Blocks to represent data using DataTable and Chart components

5 Conclusions

The constant increase in the number of mobile devices can be used to improve the processes executed in multiple areas, analysing the data produced through the incorporation of mobile applications for this purpose. This would be positively accepted by the population, as approximately half of the time spent in the digital world is directly linked to the use of mobile applications. On the other hand, the use of specific-purpose applications can generate a lot of information that can then be analysed using Learning Analytics techniques.

However, it is important to remember that the creation of specific-purpose applications is very difficult for people with limited programming skills. This work presents a methodology to help programming novices decide what type of data analysis technology to decide in each situation.

As future work, the implementation of the rest of the LA components in VEDILS will be carried out. In addition, a usability assessment will be conducted with users to evaluate the effectiveness of the proposed methodology. Finally, a virtual assistant will be implemented to provide the necessary support to follow the methodology from VEDILS.

Acknowledgements

This work has been developed in the VISAIGLE project, funded by the Spanish Ministry of Economy, Industry and Competitiveness with ref. TIN2017-85797-R.

References

1. Informe Ditrendia Mobile en España y en el Mundo 2017. https://www.amic.media/media/files/file_352_1289.pdf
2. Abowd, G.D., Dey, A.K., Brown, P.J., Davies, N., Smith, M., Steggles, P.: Towards a better understanding of context and context-awareness. In: International Symposium on Handheld and Ubiquitous Computing. pp. 304–307. Springer (1999)
3. Baker, R.S., Inventado, P.S.: Educational data mining and learning analytics. In: Learning analytics, pp. 61–75. Springer (2014)
4. Blikstein, P.: Multimodal learning analytics. In: Proceedings of the third international conference on learning analytics and knowledge. pp. 102–106. ACM (2013)

5. Campbell, J.P., Oblinger, D.G., et al.: Academic analytics. EDUCAUSE review **42**(4), 40–57 (2007)
6. Cao, L.: Data science: a comprehensive overview. ACM Computing Surveys (CSUR) **50**(3), 43 (2017)
7. Castelluccia, D., Caldarola, E.G., Boffoli, N.: Environmental big data: a systematic mapping study. ACM SIGSOFT Software Engineering Notes **41**(6), 1–4 (2017)
8. Corbi, A., Burgos, D.: Review of Current Student-Monitoring Techniques used in eLearning-Focused recommender systems and learning analytics. the experience api & lime model case study. IJIMAI **2**(7), 44–52 (2014)
9. Cruz-Benito, J., Therón, R., García-Peñalvo, F.J.: Software architectures supporting human-computer interaction analysis: A literature review. In: International Conference on Learning and Collaboration Technologies. pp. 125–136. Springer (2016)
10. Cugola, G., Margara, A.: Processing flows of information: From data stream to complex event processing. ACM Computing Surveys (CSUR) **44**(3), 15 (2012)
11. Esteban, G., Fernández, C., Conde, M.Á., García-Peñalvo, F.J.: Playing with shule: surgical haptic learning environment. In: Proceedings of the Second International Conference on Technological Ecosystems for Enhancing Multiculturality. pp. 247–253. ACM (2014)
12. Gollapudi, S.: Practical Machine Learning. Packt Publishing Ltd (2016)
13. Laurila, J.K., Gatica-Perez, D., Aad, I., Bornet, O., Do, T.M.T., Dousse, O., Eberle, J., Miettinen, M., et al.: The mobile data challenge: Big data for mobile computing research. In: Pervasive Computing. No. EPFL-CONF-192489 (2012)
14. Salber, D., Dey, A.K., Orr, R.J., Abowd, G.D.: Designing for ubiquitous computing: A case study in context sensing. Tech. rep., Georgia Institute of Technology (1999)
15. Sena, B., Allian, A.P., Nakagawa, E.Y.: Characterizing big data software architectures: a systematic mapping study. In: Proceedings of the 11th Brazilian Symposium on Software Components, Architectures, and Reuse. p. 9. ACM (2017)
16. Singh, S., Vajirkar, P., Lee, Y.: Context-based data mining using ontologies. In: International Conference on Conceptual Modeling. pp. 405–418. Springer (2003)
17. Wei, L., Zhou, H., Soe, A.K., Nahavandi, S.: Integrating Kinect and haptics for interactive STEM education in local and distributed environments. In: Advanced Intelligent Mechatronics (AIM), 2013 IEEE/ASME International Conference on. pp. 1058–1065. IEEE (2013)