

# Design and Implementation of Mobile Portal for Kenyan Pastoralists

Rashid Ali, Damaris J. N. Odero, Edna Milgo

Department of Information Technology, School of Information Sciences, Moi University, Eldoret, Kenya

Email: rash0882@yahoo.co.uk, rash8214@gmail.com, oderodjn@gmail.com, medynac@gmail.com

**How to cite this paper:** Ali, R., Odero, D.J.N. and Milgo, E. (2017) Design and Implementation of Mobile Portal for Kenyan Pastoralists. *Journal of Software Engineering and Applications*, 10, 884-890. <https://doi.org/10.4236/jsea.2017.1012050>

**Received:** November 8, 2017

**Accepted:** November 27, 2017

**Published:** November 30, 2017

Copyright © 2017 by author and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## Abstract

A mobile portal is an internet gateway that provides online services primarily accessed via mobiles and other handheld devices. Mobile portals are part of web 3.0 that is starting to revolutionize online services. Such portals can help pastoralists in Kenya access essential information and marketing services online. The aim of the study was to design and implement a mobile portal that will help Kenyan pastoralists improve their yields. The study adopted systems development life cycle (SDLC) methodology to design, implement and test the portal and used structured interviews and observation to collect user requirements. The portal was coded in J2ME, JSP and SQL languages and developed using Java Development Kit (JDK), Sun Wireless Toolkit, Lightweight User Interface Toolkit (LWUIT), Struts, Tomcat and MySQL. The outcome of the study was the mobile portal successfully developed, tested and deployed.

## Keywords

Mobile Portal, Portal, J2ME, Systems Development Life Cycle (SDLC), Pastoralists

## 1. Introduction

Kenyan pastoralists need a mobile portal that provides them with a platform to market their products and access critical information relevant to livestock management. Mobile portals and other wireless technologies have greater implications for improved livestock management [1]. They have the potential to link pastoralists to the global animal market, aid in automatic identification and tracking of animals via Radio-Frequency identification (RFID) systems and enable access to scientific knowledge on animal husbandry [2]. Many studies have highlighted problems facing the livestock sector as lack of market information

and access, gaps in animal supply chain, inadequate knowledge about proper herding, livestock diseases as well as climate change and dwindling grazing grounds [3] [4] [5]. Mobile web can help solve some of these problems facing Kenyan pastoralists.

The activities of designing and implementing the mobile portal have been guided by systems development life cycle (SDLC). SDLC is a proven methodology for developing software systems. SDLC comprises of four key steps: planning and selection, analysis, design and implementation [6]. Planning and selection is about identifying problems and selecting the right course of action [7].

## 2. System Analysis

System analysis is composed of two activities: requirements gathering and requirements analysis [8]. Requirements analysis involves techniques for structuring the information discovered during requirements determination, and they include use cases, data flow diagrams (DFD), flowcharts and decision tables, among others.

This study has used use cases and DFD level 0 to structure user requirements, a choice based on their ease of use and effectiveness in modeling user requirements.

The portal users were categorized into a number of user classes, determined by access privileges enjoyed or their occupations. Grouping by access rights yields two user groups—ordinary users and administrators—while grouping by occupation yields five users, namely: herder/trader, web master, weather system, Vet/ livestock officer and GOK/NGOs.

Two use cases and a context diagram (DFD) were used to model the requirements of these users.

### 2.1. Use Cases

A Use Case is a means of expressing user requirements; it represents how a system interacts with its environment by illustrating the activities that are performed by the users of the system and the system's responses [9].

**Figure 1** and **Figure 2** are two use cases representing the end user's and administrator's requirements respectively.

### 2.2. Data Flow Diagrams (DFDs)

A DFD is a graphical tool that allows system analysts to show flow of data in an information system; it shows the main inputs and outputs of the system and the system main users [10]. There are two types of DFDs: context level DFD (level 0) and exploded DFDs.

A context diagram is composed of external entities, the system and data flows [11]. **Figure 3** represents a DFD level 0 (context diagram) for the portal showing the five occupational users.

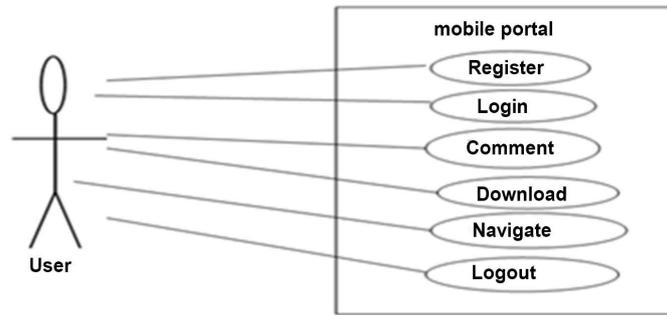


Figure 1. End user use cases.

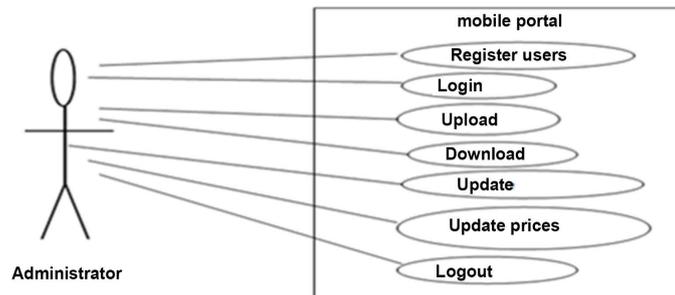


Figure 2. Administrator use cases.

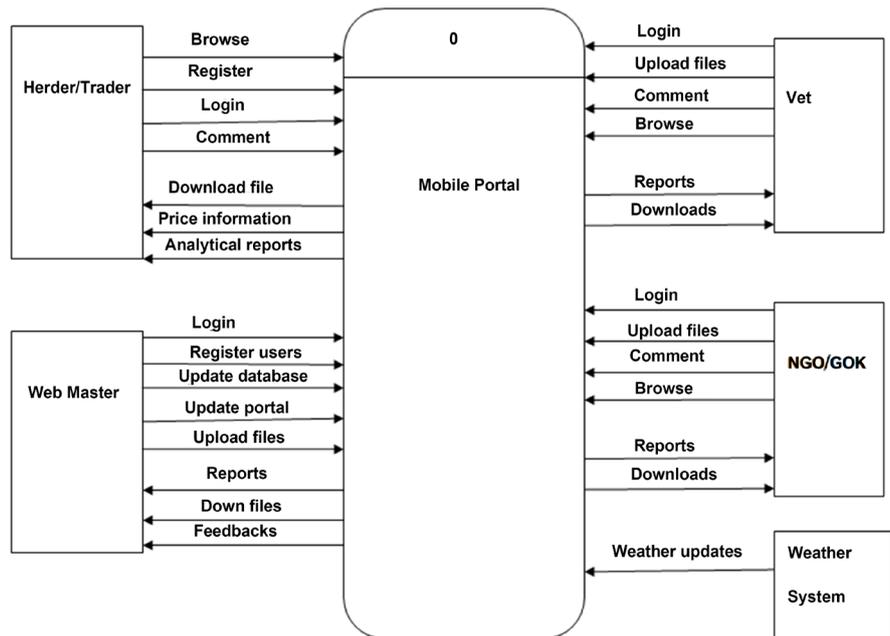


Figure 3. Context diagram for mobile portal.

### 3. System Design

System design is the determination of the overall system architecture—consisting of a set of physical processing components, hardware, software, people, and the communication among them—that will satisfy the system’s essential requirements [12]. Key system design activities include architecture design, user interface design

and database design [13].

In this section, a description of how two design activities: web application design and database design, were effectively used to design the proposed web portal is given.

### 3.1. Web Application Design

In web application design, system architecture has to be fully described and its components specified.

The mobile portal has a 3-tier architecture composed of the view/presentation layer (tier 1), the application layer (tier 2) and the data layer (tier 3), alternatively called the MVC model [14].

The presentation layer (View) is dedicated to displaying and rendering data to the user [15]. For the portal, tier 1 (presentation layer) is represented by web browsers capable of parsing HTML, CSS, JavaScript and JQuery.

Tier 2, the application layer (controller), handles processing of application logics and database accesses [16]. Apache web server (or IIS), JSP engine and tomcat represent the portal's tier 2.

Web servers parse HTTP requests from clients (the browsers), JSP engine processes JSP scripts and tomcat compiles and runs action Servlets.

Tier 3—data/model layer—consist of MySQL database management system (DBMS), database tables, stored procedures and views [17].

### 3.2. Database Design

To design the database for the portal, Entity Relationship (ER) diagram, normalized tables and views were used. Entity relationship diagram is a graphical representation of database tables [18]. The main components of ER are entities, the relationships that can exist among them and the constraints [19]. **Figure 4** is an ER diagram for the portal showing 8 tables, their columns and relationships among tables. ER is the most convenient strategy to describe data relationships and rules, and to outline key modules for the portal.

## 4. System Implementation

System implementation consists of three activities: coding, testing and documentation [20]. This section describes the coding and testing of the portal.

### 4.1. Coding

The portal was coded in JAVA, J2ME, JSP and SQL languages and developed using JDK 8, sun wireless toolkit 2.5.2, Lightweight User Interface Toolkit (LWUIT) 1.4, struts 1.3, tomcat and MySQL.

Wireless toolkit was for compiling and running midlets, MySQL for creating and managing the portal's database, tomcat for converting JSP scripts into Servlets, compiling and running action Servlets, and struts for managing communications and control between the applications and database engines.

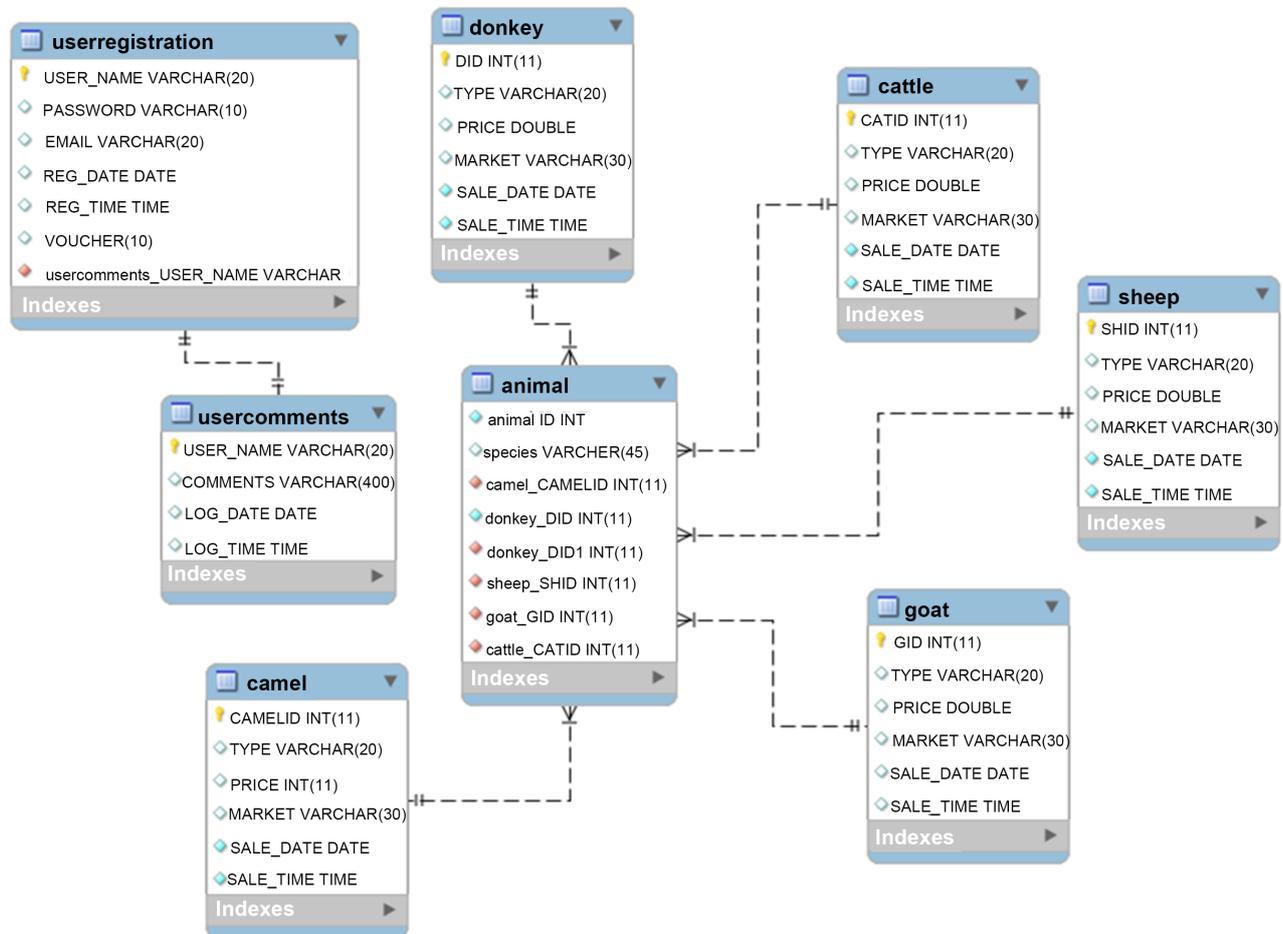


Figure 4. ER diagram for the mobile portal.

## 4.2. Testing

Two testing methods: functional testing and integration testing were used to test and examine potential vulnerabilities of the portal. In functional testing, each module is tested alone in an attempt to discover any errors that may exist in the module's code while in integration testing, modules or functions are combined together and tested collectively as a related group [21]. To functionally test modules of the portal, a number of test cases were conducted including input data validation for input forms such as user registration, login, and update price forms, testing click-ability of action buttons and workability of file upload while integration tests were run on the portal for potential bugs and usability issues such as authentication of users, the portal in different browsers and devices, examining together upload and download functionalities among others.

## 5. Conclusion

Some of the constraints faced by pastoralists against maximizing their livestock productivity arise from shortage of market information, poor information access, and lack of access to livestock insurance among others. Remedies to these problems may take a variety of forms from technological, socio-economic, poli-

cy-based etc. But on the list of viable technological solutions is a mobile web with the potential to linking the pastoralists to the global livestock market and enabling access to up-to-date market information and other needed information services. The study was therefore based on designing and implementing a workable mobile portal for Kenyan pastoralists. The portal was successfully developed and implemented.

## References

- [1] Maumbe, B.M. (2010) Mobile Agriculture in South Africa: Implementation Framework, Value-Added Services and Policy Implications. *International Journal of ICT Research and Development in Africa (IJICTRDA)*, **1**, 35-59. <https://doi.org/10.4018/jictrda.2010040103>
- [2] Guo-liana, F.A.N. and Dong-jianb, H.E. (2008) Animals Information Identifying and Tracking System Based on Electronic Identification. *Journal of Agricultural Mechanization Research*, **6**, 97-100.
- [3] Hurrissa, B. and Eshetu, J. (2002) Challenges and Opportunities of Livestock Marketing in Ethiopia. In *Proceedings of the 10th Annual Conference of the Ethiopian Society of Animal Production (ESAP)*, Addis Ababa, Ethiopia, August 2002, 1-13.
- [4] Hampshire, K. (2002) Networks of Nomads: Negotiating Access to Health Resources among Pastoralist Women in Chad. *Social Science & Medicine*, **54**, 1025-1037. [https://doi.org/10.1016/S0277-9536\(01\)00078-8](https://doi.org/10.1016/S0277-9536(01)00078-8)
- [5] Thornton, P.K., Van de Steeg, J., Notenbaert, A. and Herrero, M. (2009) The Impacts of Climate Change on Livestock and Livestock Systems in Developing Countries: A Review of What We Know and What We Need to Know. *Agricultural Systems*, **101**, 113-127. <https://doi.org/10.1016/j.agsy.2009.05.002>
- [6] Dennis, A., Wixon, B.H. and Roth, R.M. (2012) System Analysis and Design. 5th ed., John Wiley & Sons, Inc., Hoboken, NJ.
- [7] Lederer, A.L. and Sethi, V. (1988) The Implementation of Strategic Information Systems Planning Methodologies. *MIS Quarterly*, **12**, 445-446. <https://doi.org/10.2307/249212>
- [8] Shelly, G.B. and Rosenblatt, H.J. (2012) Essentials of Systems Analysis and Design. 9th ed., Course Technology, Cengage Learning, Boston, USA.
- [9] Schneider, G. and winters, J.P. (2001) Applying Use Cases: A Practical Guide. Pearson Education, India.
- [10] Chen, Y.L. (2009) Data Flow Diagram. In *Modeling and Analysis of Enterprise and Information Systems*, Springer, Berlin, Heidelberg, 85-97.
- [11] Ward, P.T. (1986) The Transformation Schema: An Extension of the Data Flow Diagram to Represent Control and Timing. *IEEE Transactions on Software Engineering*, **2**, 198-210. <https://doi.org/10.1109/TSE.1986.6312936>
- [12] Börger, E. (1998) High Level System Design and Analysis Using Abstract State Machines. In *Proceedings of International Workshop on Current Trends in Applied Formal Methods*, Springer, London, 1-43.
- [13] Hoffer, J.A. (2012) Modern Systems Analysis and Design. 6th ed., Pearson Education, India.
- [14] Selfa, D.M., Carrillo, M. and Boone, M.D.R. (2006) A Database and Web Application Based on MVC Architecture. In *Electronics, Communications and Computers, 2006. CONIELECOMP 2006. 16th International Conference on IEEE*, Puebla, Mex-

- ico, 1 March-27 February 2006, 48. <https://doi.org/10.1109/CONIELECOMP.2006.6>
- [15] Deacon, J. (2009) Model-View-Controller (MVC) Architecture. Online [Citado em: 10 de março de 2006.] <http://www.jdl.co.uk/briefings/MVC.pdf>
- [16] Sanderson, S. (2008) Architecture. Apress, New York, 51-99. [https://doi.org/10.1007/978-1-4302-1647-6\\_3](https://doi.org/10.1007/978-1-4302-1647-6_3)
- [17] Leff, A. and Rayfield, J.T. (2001) Web-Application Development Using the Model/View/Controller Design Pattern. In *Enterprise Distributed Object Computing Conference, 2001. EDOC'01. Proceedings Fifth IEEE International*, IEEE, Seattle, WA, USA, 4-7 Sept. 2001, 118-127. <https://doi.org/10.1109/EDOC.2001.950428>
- [18] Thalheim, B. (2013) Entity-Relationship Modeling: Foundations of Database Technology. Springer Science & Business Media, Berlin.
- [19] Isakowitz, T., Stohr, E.A. and Balasubramanian, P. (1995) RMM: A Methodology for Structured Hypermedia Design. *Communications of the ACM*, **38**, 34-44. <https://doi.org/10.1145/208344.208346>
- [20] Tilley, T., Cole, R., Becker, P. and Eklund, P. (2005) A Survey of Formal Concept Analysis Support for Software Engineering Activities. *Formal Concept Analysis*, **3626**, 250-271. [https://doi.org/10.1007/11528784\\_13](https://doi.org/10.1007/11528784_13)
- [21] Kuhn, D.R. and Reilly, M.J. (2002) An Investigation of the Applicability of Design of Experiments to Software Testing. In *Software Engineering Workshop, 2002. Proceedings 27th Annual NASA Goddard/IEEE*, IEEE, Greenbelt, MD, USA, 4-6 December 2002, 91-95. <https://doi.org/10.1109/SEW.2002.1199454>