Journal of Engineering and Applied Sciences 12 (10): 2762-2769, 2017
ISSN: 1816-949X
© Medwell Journals, 2017

Feature Driven Development of a Smartphone Based Vision-Aware mHealth Framework

[1]Melvin Cabatuan, [1]Elmer Dadios, [1]Laurence Gan Lim, [1]Jose Santos Carandang VI, [1]Eric Punzalan, [1]Ma. Luisa Enriquez, [1]Dennis Erasga, [1]Michael Ples, [1]Romeo Teruel, [1]Balintawak Sison-Gareza, [1]Ma. Ellenita De Castro, [1]Julien Carandang and [2]Raouf Naguib
[1]De La Salle University, 1004 Manila, Philippines
[2]BIOCORE Research and Consultancy International and Liverpool Hope University, UK

Abstract: Mobile health (mHealth) applications attempt to capitalize on the ubiquity and exponential growth of mobile technologies for the benefit of public health, leading to a growing research interest in devising frameworks for addressing specific or general mHealth challenges. In this context, the primary goal of this study is to present a novel smartphone-based development framework for prototyping vision-aware native mHealth applications developed using Feature Driven Development (FDD) methodology. Then, we describe a prototype mHealth educational application, 'Dibdib'** Advocacy App for breast cancer awareness, utilizing the proposed vision-aware mHealth framework in Android platform. The results illustrate that FDD is feasible option in mHealth application development under the proposed mHealth framework.

Key words: mHealth, Feature Driven Development (FDD), computer vision, breast cancer awareness, framework

INTRODUCTION

Mobile health is defined by the World Health Organization as "the practice of medical and public health through the usage of mobile devices". Recently there are over 7 billion wireless subscribers worldwide according to the International Telecommunication Union corresponding to a penetration rate of 97%. This scenario offers an enormous potential for positive impact in the healthcare sector especially in low and middle income countries leading to an increasing trend of mHealth application development. Among the principal aims of an application development framework in mHealth is to provide an abstraction layer for the developer and aid in dealing with the complicated low level programming details involved in the core of a mobile application. This approach can be valuable for simplifying or expediting the mHealth application development. Many relevant advances have been proposed previously, namely: android open platform with multi-interface biomedical modules for physiological measurement a mobile phone platform for collecting psychological, physiologica land activity information for mental health research; android open-source smartphone framework for sensors data acquisition, signal processing, pattern analysis, interaction and feedback open-source programming framework for prototyping and management of wireless body sensor networks applications (Fortino *et al*., 2013); web-based breast cancer risk assessment mHealth platform (Ghaderi *et al*., 2015) and an mHealth framework which supports resource and communication abstraction, biomedical data acquisition, health knowledge extraction, persistent data storage, adaptive visualization, system management and value-added services (Banos *et al*., 2015). However, to the best of our knowledge, mHealth framework that specializes in smartphone computer vision based functionality has not yet been proposed. Among, the main advantages of the proposed vision-aware approach is its unobtrusiveness to the user and ubiquity since most smartphone already incorporates built-in camera sensor.

MATERIALS AND METHODS

In the development of the proposed vision-aware mHealth framework we utilized one of the principal agile methodologies-Feature-Driven Development (FDD) as shown in Fig. 1. FDD was originally presented as the solution to the problem of accommodating shorter business cycles which may deemed as appropriate for the mHealth sector. FDD is a model-driven, short-iteration process which begins with establishing an overall domain model from consultation with domain experts then it continues with a series of two-week "design by feature, build by feature" (DBF-BBF) iterations (Palmer and Felsing, 2001). The FDD methodology starts by building

Corresponding Author: Melvin Cabatuan, De La Salle University, 1004 Manila, Philippines

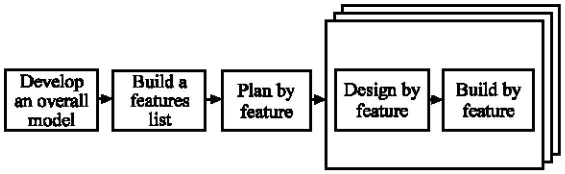

Fig. 1: Feature Driven Development (FDD) is composed of five core processes, namely: develop an overall model; build a features list; plan by feature; design by feature and build by feature. The last two processes, "design by feature" and "build by feature" are iterative sequences which are composed of designing, building and testing features in at most 2 weeks time period

class diagrams depicting the significant types of objects within a particular problem domain and the relationships between them (Palmer and Felsing, 2001). Initial tasks include formation of a modeling team, domain walk-through, document (or literature) review, model development, overall object model refinement and model notes writing. Furthermore, we also added the formulation of the general system architecture in the initial FDD process. Secondly, a features list is built. An atomic feature is a tiny building block for planning, reporting and tracking which is user-valued and understandable, measurable and do-able with at most 2 weeks completion time. In this process we also employed a use case diagram to gain an understanding of the big picture as they relate to the features. Thirdly, planning by feature involves creating a plan which mainly consists of expected completion dates, class ownership and feature assignments. Finally, the FDD process is concluded by a series of Design and Build by Feature (DBF/BBF) iteration with 6 milestones namely: domain walkthrough (1%); design (40%); design inspection (3%); coding/testing (45%); code inspection (10%) and promote to build (1%). The corresponding percentages are recommended completion time with respect to the total feature completion schedule.

RESULTS AND DISCUSSION

Among the results of our FDD Process 1 (develop an overall model) implementation is the general smartphone-based mHealth architecture (Fig. 2). Based on consultation and meetings with domain experts, i.e., physicians, health informatics experts, medical researchers and practitioners and users, together with community field visits and extensive literature review of existing mHealth systems and applications we proposed a customizable, service-oriented, feature-based mHealth framework. Public health systems in general can be categorized into 4 types of care, namely: promotion; prevention; intervention and diagnosis with various services as health promotion, self-care, early intervention, assistive living, consultation and counseling (Campbell *et al*., 2014). Furthermore, each of the health services (except in diagnosis) may involve the following common features, namely: education, monitoring, management, communication and support (Campbell *et al*., 2014). Based on these features we formulated an mHealth framework which is an aggregate of independent microservices in which any combination comprises a tailored mHealth system for a specific problem domain.

Figure 3 illustrates the simplified overall mHealth domain model where the mHealth App is a composition of a specific problem domain, user-interface, system interaction, data services and utilities. Figure 4 shows the major mHealth framework use cases which are defined as system activities that help accomplish some goal of the user of the system (Satzinger *et al*., 2011). The mHealth application is depicted as a nonhuman actor. Accordingly, the five major mHealth framework use cases corresponds to the major features, i.e., education, monitoring, management, communication and support that may be supported by any mHealth app that provides either health promotion, disease prevention and/or intervention. It should be noted that diagnosis is excluded in the supported mHealth framework services as this type of care is much more complicated and may involve a series of tests and essential face-to-face interaction with a physician.

After determining the overall mHealth framework domain model, the next step is developing a feature list and planning or scheduling its development. In FDD a feature is a tiny building block for planning, reporting and tracking which is user-valued, understandable by the client, specific, measurable and do-able in at most two weeks. Thus, we decided that this phase and the next phases of the FDD process should be tailored to a particular prototype mHealth application to test the capabilities of the proposed mHealth framework and to limit the scope of the 1st iteration of the whole FDD process.

A prototype smartphone mHealth application called dibdib advocacy app or simply "Dibdib App" was developed utilizing our proposed mHealth framework. The main goal of Dibdib App is spreading awareness and helping filipino women fight against the breast cancer disease with general education, information gui-dance and regular breast examination tools based on the smartphone's built-in camera. It should be noted however that the intended purpose of Dibdib App is generally

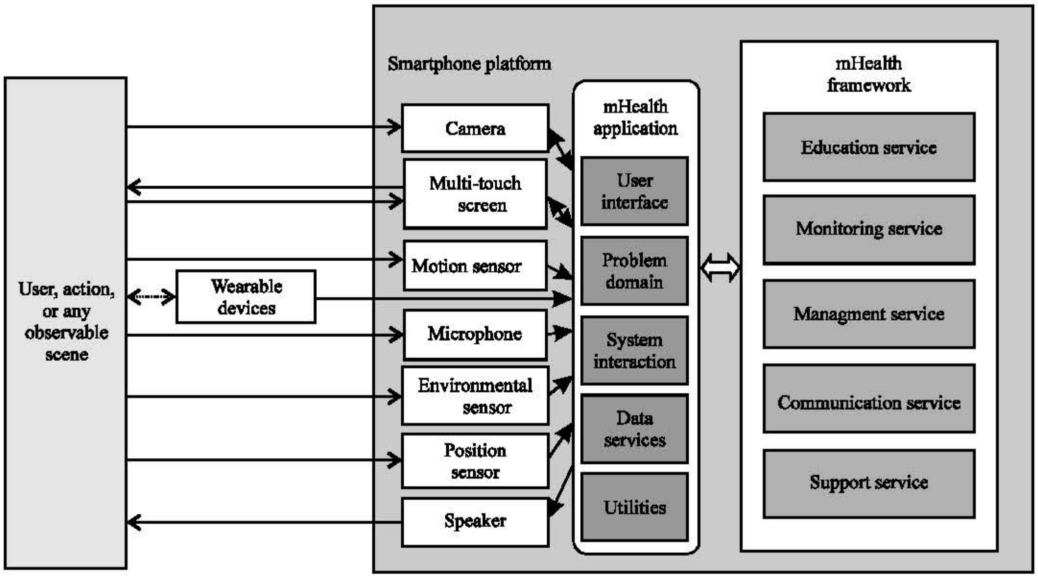

Fig. 2: Proposed smartphone-based mhealth architecture which includes the user performing an observable physical activity, smartphone sensors, mhealth application and the mhealth framework composed of education, monitoring, management, communications and support services

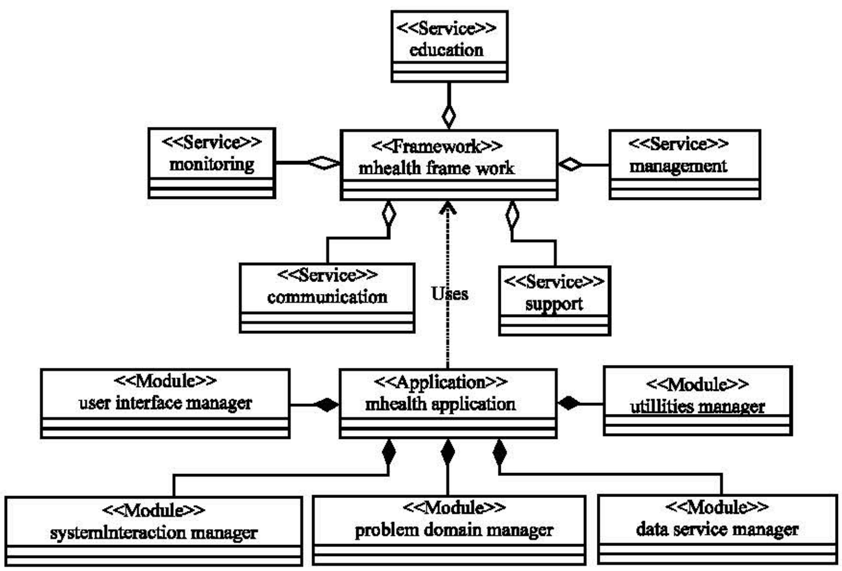

Fig. 3: A simplified overall mHealth domain model

educational or Breast Self-Examination (BSE) user assistance rather than diagnosis of breast cancer as thisis not supported by the proposed mHealth framework.

Breast cancer is the most common cancer affecting women worldwide. Philippines is among the countries with highest trends in incidence of the disease, according to the Globocan research data. Majority of deaths occur in low and middle-income countries where most women are diagnosed in late stages due mainly to lack of awareness on early detection and barriers to health services. Thus, public education is a key step because early detection cannot be successful if the public is unaware of the problem or has adverse misconceptions about the value of early detection (Ozmen and Anderson, 2008). With this in mind, eight core objectives was formulated as shown in Fig. 5. Taking the core objectives into account and its domain model, Dibdib App's feature list is established as shown in Fig. 6. Some of the features

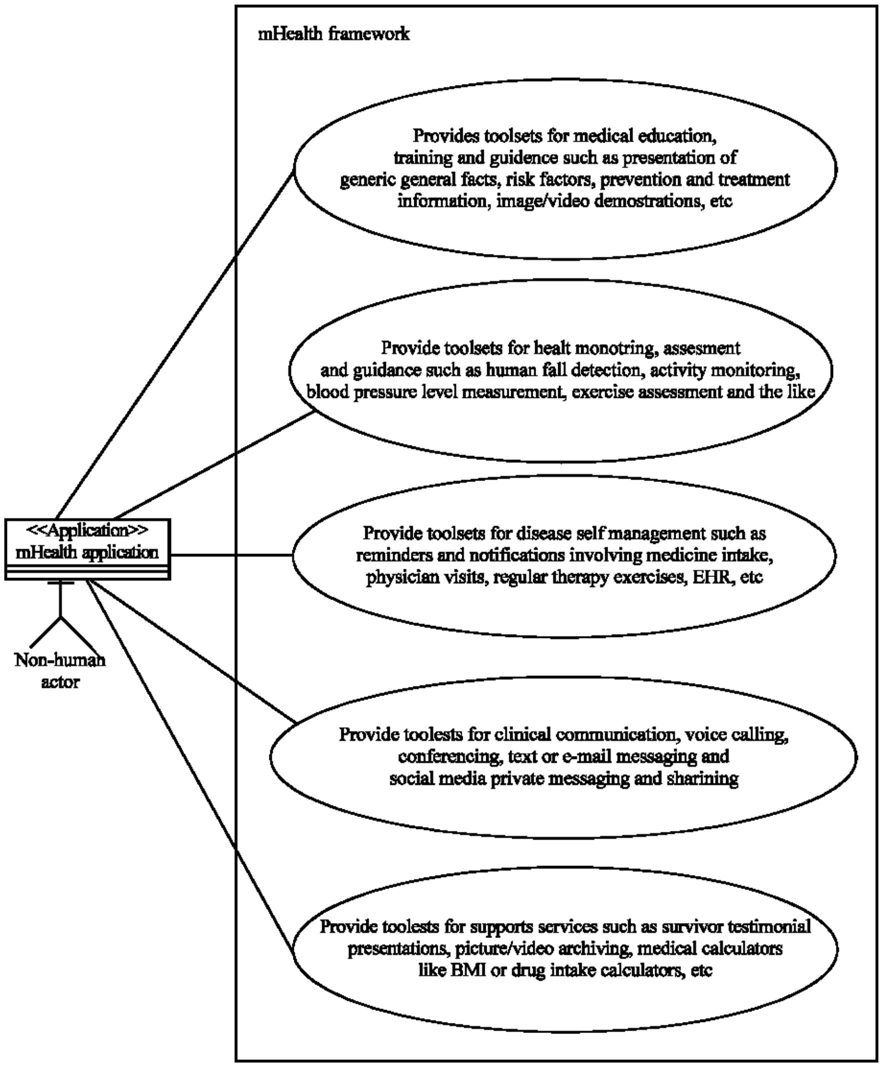

Fig. 4: mHealth framework use case diagram

can be attributed to the app's user-interface, system interaction, data services and utilities while other features cluster to mHealth framework services. Education service readily covers the presentation of basic cancer facts, risk factors, lower risk, visual check and palpation information and video playback. Both education and monitoring services accounts for camera-guided BSE performance, assessment and recording. Finally, Support service covers the map viewing of medical facilities in the locality of the user. Management service handles the recording of breast abnormality location and description. Noticeably, this FDD process iteration of our Dibdib App does not yet include communication service which is allotted for the next FDD iteration (Fig. 5).

Figure 6 depicts the use case diagram of Dibdib App under the vision-aware mHealth platform as a result of FDD overall model development process. Firstly, the principal actors of the Dibdib App system where identified as guest, registered user, medical practitioner and vision module manager (member of Monitor service). Secondly, the goals or tasks performed by each actor towards the system were determined. A registered user role can perform almost all functionalities of the system, i.e., view fact items consisting of basic breast cancer facts, common risk factors, tips on lowering the risk and BSE visual inspection guide; perform and record camera-guided BSE; record breast abnormality location and description; view BSE and its assessment with breast abnormality record (if any); search and map location of nearest medical facility and view help and support information. However, the user cannot add, modify or delete fact items nor assess the BSE performance which

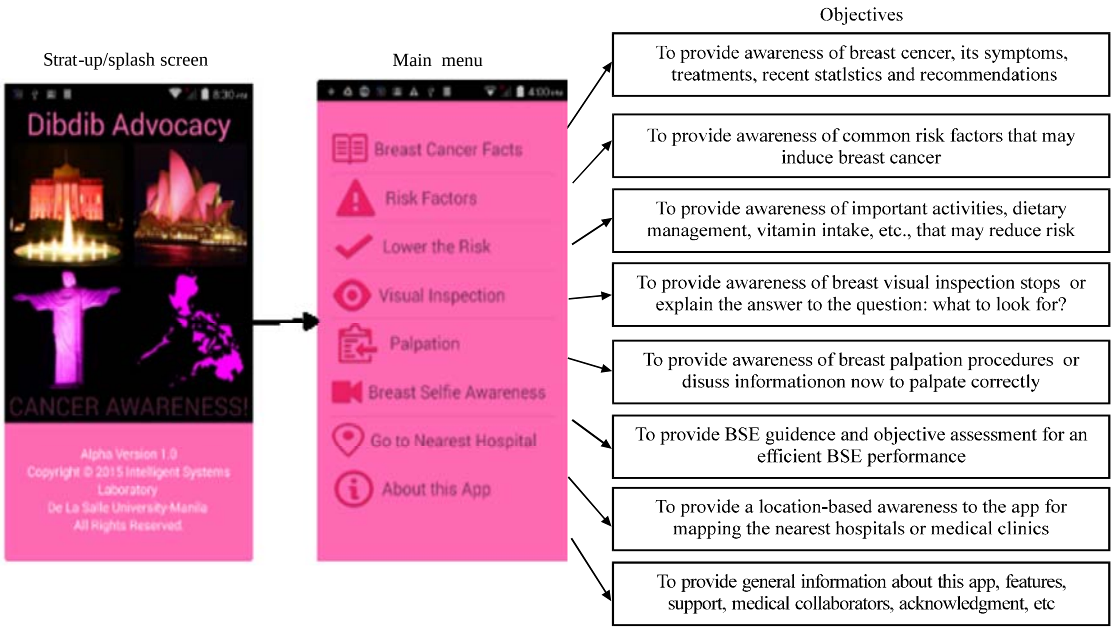

Fig. 5: Dibdib App startup splash screen; main menu composed of breast cancer facts, risk factors, lower the risk, visual inspection, palpation, breast selfie awareness, go to nearest hospital and about this app and 8 core objectives developed through consultation with domain experts and literature survey

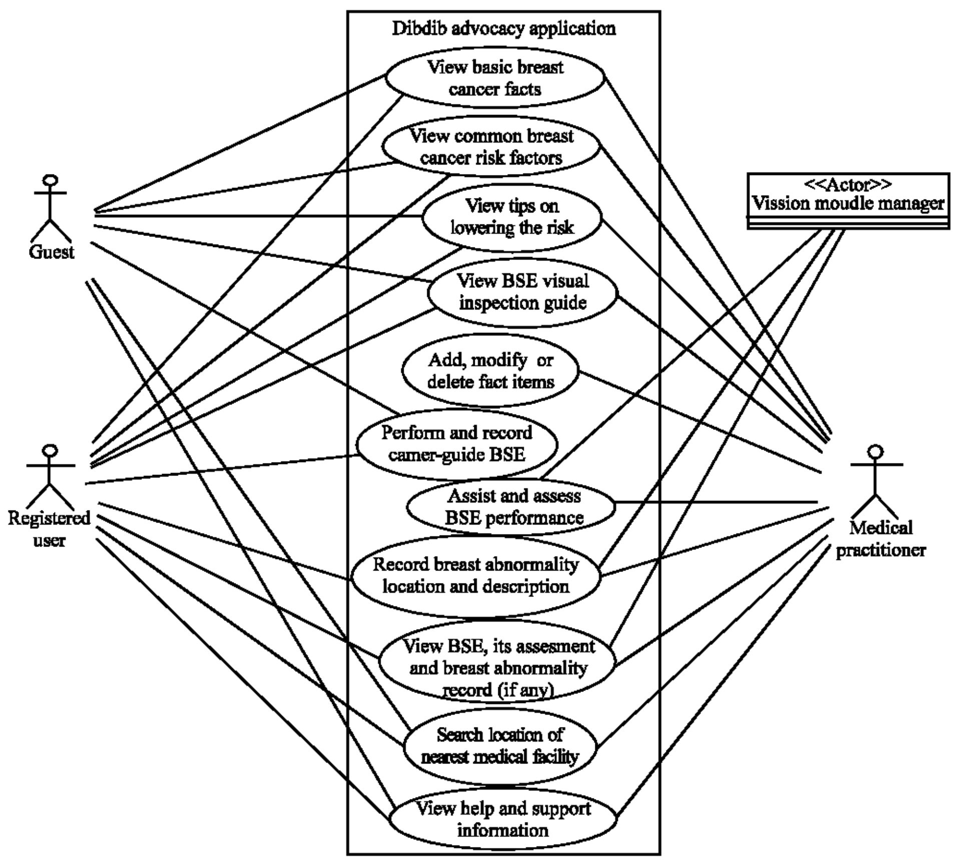

Fig. 6: Dibdib App use case diagram

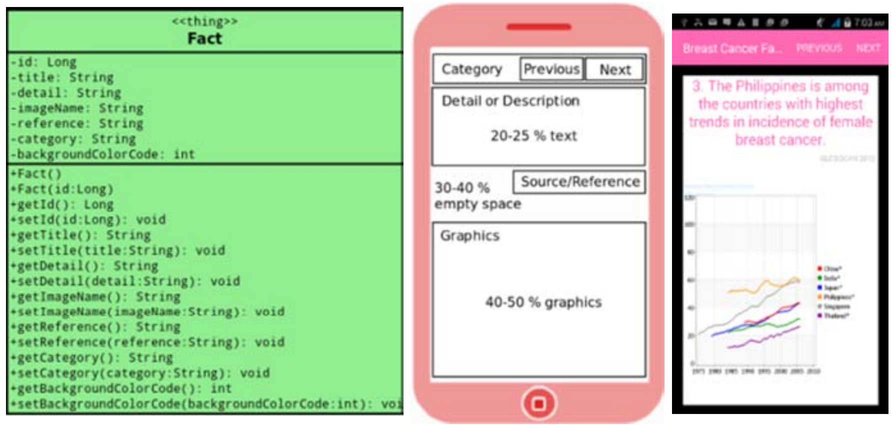

Fig. 7: Fact UML class diagram (left), fact detail activity design (middle) and its actual implementation in dibdib app (right)

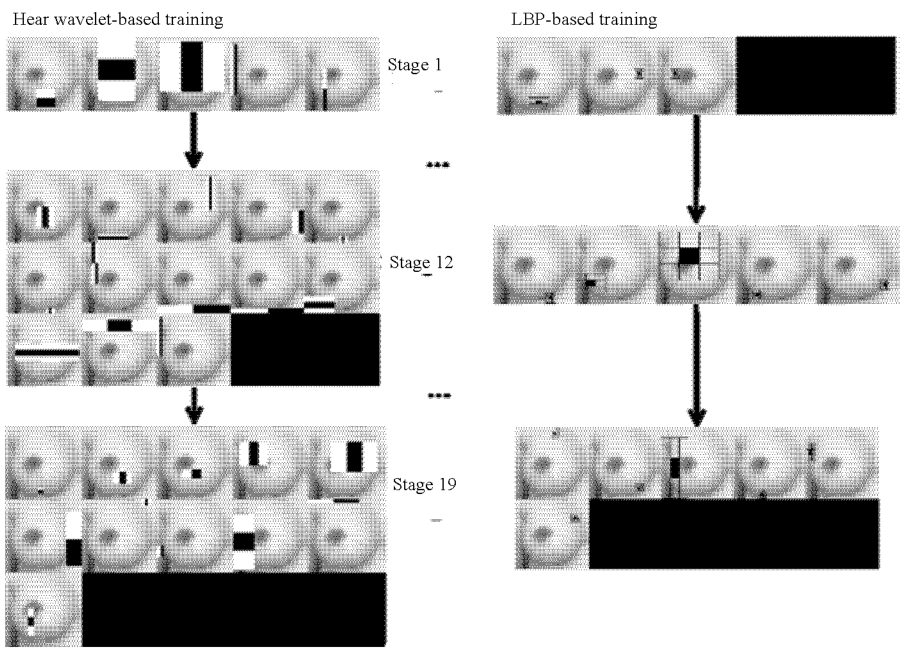

Fig. 8: Comparison between haar wavelet-based and LBP-based cascade breast detection training features

will be done by the medical practitioner role. The medical practitioner role can execute almost all capabilities of the user except the performance and recording of camera-guided BSE. The guest role also has capabilities similar to the user except the recording of breast ab-normality location and description. Finally, the vision module role (nonhuman actor) is to assist and assess BSE performance, record breast abnormality location and description and view BSE and its assessment with breast abnormality record (if any).

In poster design, the International Association of Clinical Research Nurses (IACRN) suggests a balance black and white spaces consisting of 20-25% text, 40-50% graphics and 30-40% empty space. Following a similar approach, the Dibdib App fact-based detail presentation activity was designed as shown in the middle of Fig. 7 and 8 implemented in the rightmost screen capture. Core module management is responsible for the fact detail activity presentation in which it utilizes a view pager object to present the fact objects in a simple swipe view. Furthermore, the data services module is responsible for management and persistence of fact objects. The data services module uses a SQLite database for persistent storage and query of fact objects during startup of the

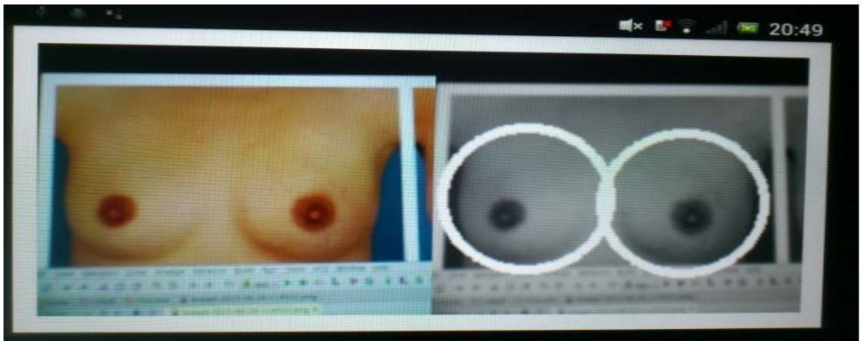

Fig. 9: Dibdib App breast detection with LBP cascade classifier

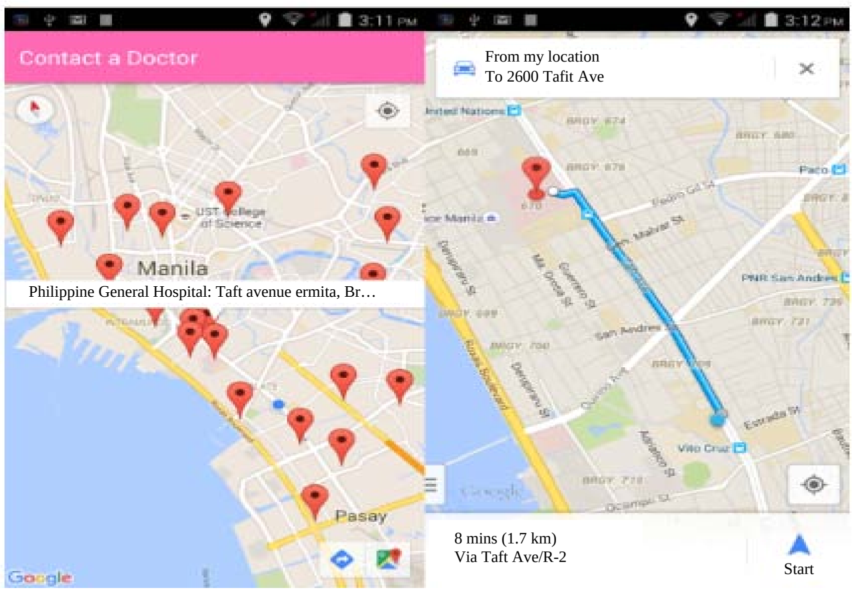

Fig. 10: Dibdib App Google places integration to locate the nearest medical facility

application to populate the top-level menu titles and detail activity items. For breast detection we employed the widely known Viola and Jones algorithm in open Computer Vision (open CV) which is an open-source and highly-efficient library for computer vision. The dataset B1000 utilized in the algorithm development is composed o f 1120 frontal breast images which is an augmentation of datasets from studies (Mohammadi *et al*., 2014) and T100 testing images composed of 100 selected image frames from BSE videos produced under the commission on higher education-philippine higher education research network (CHED-PHER net). In B1000 dataset, there are 2072 ground truth values, comprised of the bounding boxes for the left and right breasts, annotated using the standard annotation tool that comes with open CV. Consequently, the ground truth values (positive) were parsed into the open CV data vector then applied to open CV cascade classification training tool which produces the object model in an xml file. Among the most important training parameter is the type of feature to apply, e.x. Haar wavelet or Local Binary Pattern (LBP) features. Haar wavelet-based object models are reported to provide higher accuracy but LBP features is faster in training and detection due to integer calculations instead of floating point calculations in Haar wavelet-based models (Puttemans *et al*., 2015). In Dibdib App development we trained in both LBP and Haar wavelet-based model as shown in Fig. 8. Haar wavelet-based model took >94% more time than the LBP model but with lower accuracy at 70% on initial testing while the LBP model is at about 91% as tested on T 100 dataset a test set composed of randomly selected BSE images taken from BSE video frames. Figure 9 shows a sample screenshot of Dibdib App camera-based breast detection feature. Additionally, Fig. 10 shows the Google places API integration of the Dibdib App.

CONCLUSION

The main contributions of this study were as follows: a proposed novel smartphone-based framework for vision-aware mHealth applications; sample implementation of FDD in smartphone application development and presentation of an example prototype application, Diddib App for breast cancer awareness, BSE guidance and assessment as a proof of concept. For future research, more testing are needed, e.g., usability and App quality assessment. Furthermore, some app features were not discussed due to the limited scope of this study.

ACKNOWLEDGEMENT

This project is financially supported by the Commission of Higher Education (CHED)-Philippine Higher Education Research Network (PHERNet) Sustainability Studies Program.

REFERENCES

Banos, O., C. Villalonga, R. Garcia, A. Saez and M. Damas *et al*., 2015. Design, implementation and validation of a novel open framework for agile development of mobile health applications. Biomed. Eng. Online, 14: S2-S6.

Campbell, D., E.G. Pereira and G. McDowell, 2014. Ontology driven framework for personal mhealth application development. Proceedings of the 2014 8th International Conference on Next Generation Mobile Apps Services and Technologies (NGMAST), September 10-12, 2014, IEEE, Ormskirk, England, ISBN:978-1-4799-5074-4, pp: 320-325.

Fortino, G., R. Giannantonio, R. Gravina, P. Kuryloski and R. Jafari, 2013. Enabling effective programming and flexible management of efficient body sensor network applications. IEEE. Trans. Hum. Mach. Syst., 43: 115-133.

Ghaderi, M., G. Gupta, L. Tamil, M. Abdallah and K. Qaraqe, 2015. mHealth platform for breast cancer risk assessment. Proceedings of the 2015 International Conference on Healthcare Informatics (ICHI), October 21-23, 2015, IEEE, Dallas, Texas, ISBN:978-1-4673-9549-6, pp: 570-573.

Mohammadi, E., E.P. Dadios, L.A.G. Lim, M.K. Cabatuan and R.N. Naguib *et al*., 2014. Real-time evaluation of breast self-examination using computer vision. J. Biomed. Imaging, 2014: 1-16.

Ozmen, V. and B.O. Anderson, 2008. The challenge of breast cancer in low-and middle-income countries-implementing the breast health global initiative guidelines. US Oncol. Touch Briefing, 76: 31-34.

Palmer, S.R. and M. Felsing, 2001. A Practical Guide to Feature-Driven Development. Pearson Education, Upper Saddle River, New Jersey, ISBN:0130676152,.

Puttemans, S., J. Howse, Q. Hua and U. Sinha, 2015. OpenCV 3 Blueprints: Expand your Knowledge of Computer Vision by Building Amazing Projects with OpenCV 3. Packt, Birmingham, UK., ISBN:9781784399757,.

Satzinger, J.W., R.B. Jackson and S.D. Burd, 2011. Systems Analysis and Design in a Changing World. 6th Edn., Cengage Learning, Boston, Massachusetts, ISBN-13:978-1-111-53415-8, Pages: 486.