MOBILE PHONE APPLICATION PROGRAMMING INTERFACES FOR E-COMMERCE

Thomas Mugo[1], Karari Ephantus Kinyanjui[2] and Kihuha Cyrus Kamau[3]

[1]Department of Computer Science, Dedan Kimathi University of Technology, Nyeri, Kenya

`mugo.thomas@ibearesearch.org`

[2]Department of Computer Science, Dedan Kimathi University of Technology, Nyeri, Kenya

`efantus.kinyanjui@dkut.ac.ke`

[3]Department of Computer Science, Dedan Kimathi University of Technology, Nyeri, Kenya

`kamau.cyrus@gmail.com`

ABSTRACT

As the mobile and Internet penetration rates in Kenya rise every year, online business and e-commerce will steadily pick. This will have a huge impact on how we live and how we transact business. However, a few more things have to be in place before we can reach that time when e-Commerce will be commonplace to all individuals. Payments, security and merchandise delivery solutions will have to be developed to facilitate smooth e-commerce. This paper is an extract from a project report on the issue of payment in a Kenyan e-commerce environment.

KEYWORDS

Kenya, Mobile, Payment, e-Commerce, API

1. INTRODUCTION

Besides having a good internet connection and an e-commerce portal, a customer making purchases of a good or a service electronically will want to make payments in a secure, seamless and unobtrusive way.

The payment process should be brief, easy to understand, not suspicious and most of all seamless on the customers end while for merchants and retailers it should be easy to integrate and manage. The current solutions fail to meet the threshold of the afore mentioned conditions for the Kenyan market while others like use credit/debit cards have caught little traction in the Kenyan market.

2. PROJECT OBJECTIVES

2.1. Main objective

The main objective of the study was to come up with a cross platform mobile payments Application Programming Interface (API).

2.2. Specific objectives

The API will have to be a RESTful API to enhance easy integration and facilitate payments across electronic commerce platforms.

3. PROJECT JUSTIFICATION

With a mobile phone penetration rate standing at over 63% and internet penetration rate at over 20% and doubling almost annually for the past two years, there is absolutely no doubt that electronic business tractions can be conducted conveniently and by the masses if mobile phone payments technology is harnessed and applied to online transactions and other forms of electronic transactions.

This is further backed by the impressive reception that mobile payments have received in the Kenyan market. A RESTful mobile payments API would ensure that merchants and retailers can build e-commerce portals and be able to accept mobile payments from various platforms that their applications were built on.

4. GENERAL SCOPE AND APPLICATION OF THE PROJECT

The mobile payments API would exist as a third party service relying on the existing mobile payments solution by telecommunication service provider such as M-Pesa. This service will stand in between e-commerce applications and the main mobile payments service provider thus facilitating customers to seamlessly transact across the applications and the main service.

5. LITERATURE REVIEW

There are two other mobile payments APIs that have been built for the Kenyan mobile payments market; PesaPI and PasaPal. The two APIs are different and have different weaknesses. Pesapi is an open source Application Programming Interface designed for MPesa. Pesapal is a proprietary API that works by partnering with Banks, Mobile Network Operators and Credit Card companies to give consumers as many payment options as possible.

The details of these APIs can be accessed via internet with links provided in the references section here-in.

The two APIs are not platform independent and their integration with e-commerce applications is not easy.

Other methods in use in other markets for instance PayPal and Moneybookers are heavily dependent on credit/debit cards which are not very popular in the country.

Harnessing the advantages of high penetration and positive reception that mobile payments have had in Kenya requires a familiar and acceptable solution to online and electronic payments to be developed. The strengths of this API will be in its ease of integration and ability to be used across platforms.

7. METHODOLOGY

The API will be based on REST (Representational State Transfer) style of software architecture. The API will therefore make use of the following standards based on REST:

a. HTTP
b. URL
c. JSON (Resource Representations)
d. text/xml, text/html, text/json

7.1. Resources

7.1.1. Hardware

1. Linux Server box
2. Mobile equipment e.g. phone, modem
3. Computer terminals e.g. laptops and desktops

7.1.2. Software

1. Debian based Linux distribution - Ubuntu
2. Database server - Postgresql
3. SMS gateway – Kernel
4. Python web and API frameworks – django, piston, tastypie

7.2. System development approach

The System was developed in short sprint sessions with one major object of development to be tackled. As a result, the scrum methodology was opted for development.

Scrum is an iterative and incremental agile software development method for managing software projects and product or application development.

8. ANALYSIS AND DESIGN

8.1. System design

To satisfy the requirements that have already been specified and set beforehand, we went for the Object-Oriented Analysis and Design (OOAD) approach for this application. With this approach, we would be able to view the system as a group of interacting objects.

Each object represents some entity of interest in the system being modeled, and is characterized by the class, the state (data elements), and the behavior. The following diagrams define the architecture, components, and modules of the API under development.
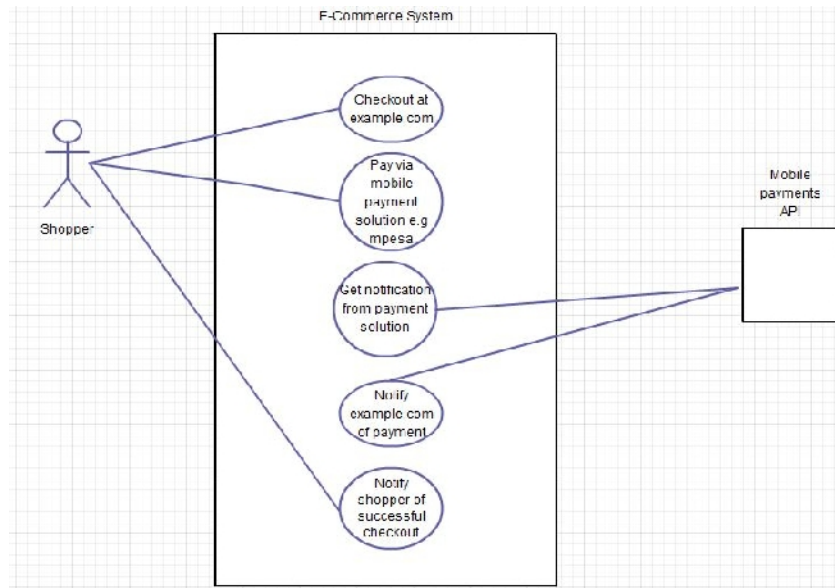
8.1.1.Use case diagram

Figure 1: Use case diagram

The API's intended clients/users are e-commerce applications/modules.
The system process in prose happens as follows:

- A user initiates a checkout from an e-commerce application and chooses a mobile payment solution as the preferred method of payment.
- The user then makes the payment using their phone.
- The API receives immediate payment notification from the payment solution of the payment.
- The API notifies the e-commerce application of successful payment and hands over the flow back to the application.
- The e-commerce application finalizes the transaction and notifies the user of success.
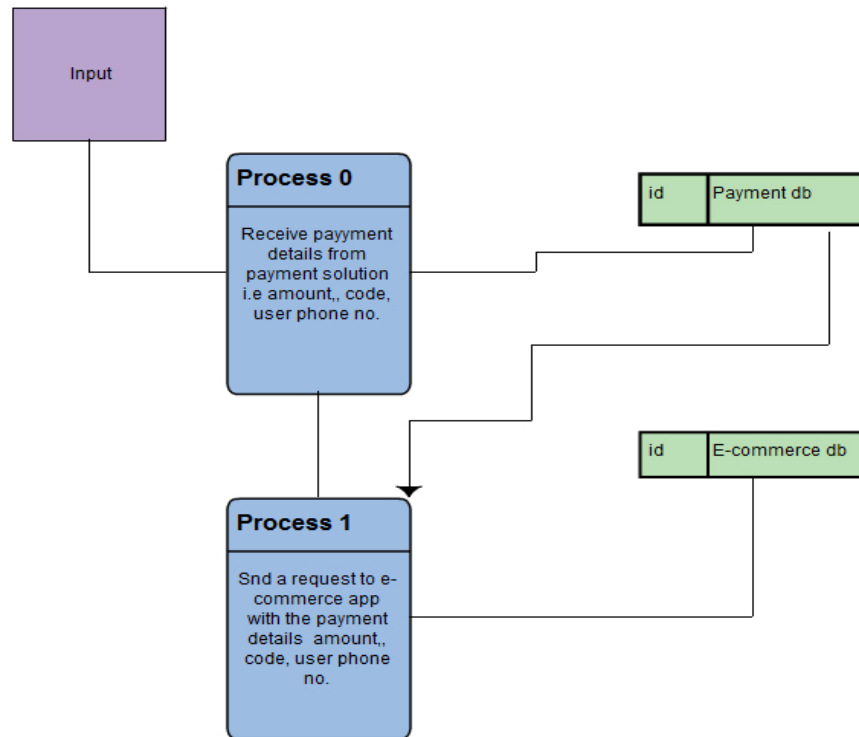
8.1.2.Dataflow diagram

Figure 2: Dataflow diagram

8.2. API resources

Resource endpoint: http://127.0.0.1:8000/api/v1/payment/"

Data parameters: '{"user_id":1, "code":"sdgsdg", "amount":80, "product_code":"sdgdgs"}'

Example request using curl:

curl -i -H "Content-Type: application/json" -X POST --data '{"user_id":1, "code":"sdgsdg", "amount":80, "product_code":"sdgdgs"}' http://127.0.0.1:8000/api/v1/payment/

Underlying database models

class Payment(models.Model):

  user = models.ForeignKey(User)

  code = models.CharField(max_length=32)

  amount = models.DecimalField(max_digits=12, decimal_places=3,

                 null=True, blank=True)

  product_code = models.CharField(max_length=32)

  created_at = models.DateTimeField(auto_now_add=True)

8.3. Challenges

Development and implementation of this API faced the following challenges:

a. Technology – Several different technologies had to come together and work in harmony to get the API working.
b. Resources – For testing purposes, limited resources were available and it was difficult to create a near production environment thus simulations had to be used.

9. CONCLUSION AND RECOMMENDATION

9.1. Limitation

The most outstanding limitation is in getting testing sandboxes from the payment solutions provides. Getting pay-bill numbers and till numbers for testing is very challenging due to long and demanding due process put in place by the providers.

9.2. Recommendation

While this system is built to be generic, and to work with various payment notification solutions provided by payment solutions, We would recommend that http post notifications be applied whenever possible as it makes use of a reliable network (internet) and reliable networking protocols.

9.3. Conclusion

The objectives of building this API have been sufficiently achieved. Further testing and more intensive integration would be needed for the system to be certified as robust enough to be used in a production environment.

With this API, business and organizations can now leverage it to receive payments via popular mobile payment systems in the market and be able to provide the purchased goods and services on an immediate basis.

REFERENCES

[1] http://mobilemonday.co.ke/2011/kenyan-mobile-phone-penetration-is-now-over-63/   - Date of last access – 7/2/2012
[2] http://en.wikipedia.org/wiki/List_of_countries_by_number_of_Internet_users   - Date of last access – 7/2/2012
[3] http://www.techmtaa.com/2011/10/05/cck-quarterly-report-98-of-internet-users-in-kenya-access-through-mobile/   - Date of last access – 7/2/2012
[4] http://tomayko.com/writings/rest-to-my-wife   - Date of last access – 7/2/2012
[5] http://www.xfront.com/REST-Web-Services.html   - Date of last access – 7/2/2012

[6]   http://www.xfront.com/REST-Web-Services.html   -   Date of last access – 7/2/2012
[7]   http://en.wikipedia.org/wiki/Representational_state_transfer   -   Date of last access – 7/2/2012
[8]   http://developer.pesapal.com/forum/2-pesapal-integration/172-pesapal-api-reference-Date of last access-7/2/2012
[9]   http://blog.komzinc.com/2012/02/pesapi-m-pesa-api.html-PasaPi Mpesa API- 21/8/2012

AUTHORS

Name: Thomas T. Mugo

Mobile: +254 724 614 314
Interest: Software development, social media
Occupation: Researcher and application developer, IBEA Research,
www.ibeareseach.org

Name: Ephantus Kinyanjui Karari
Mobile: +254 721 374 616
Interest: Web and Mobile computing
Occupation: Lecturer & PhD Student; Department of Computer Science,
KUT (www.dkut.ac.ke), Nyeri, Kenya

Name: Kahuha Cyrus Kamau

Mobile: +254 722 808 500
Interest: Web and Mobile computing
Occupation: Lecturer , Department of Computer Science,
DKUT (www.dkut.ac.ke), Nyeri, Kenya