DigiTally: Piloting Offline Payments for Phones

Khaled Baqer
Computer Laboratory,
University of Cambridge
Cambridge, UK
Khaled.Baqer@
cl.cam.ac.uk

Ross Anderson
Computer Laboratory,
University of Cambridge
Cambridge, UK
Ross.Anderson@
cl.cam.ac.uk

Jeunese Adrienne Payne
Independent Consultant
Cambridge, UK
jp@jeunesepayne.com

Lorna Mutegi
Strathmore University
Nairobi, Kenya
lmutegi@strathmore.edu

Joseph Sevilla
Strathmore University
Nairobi, Kenya
joe@strathmore.edu

ABSTRACT

Mobile payments support a range of services in many less developed countries including everyday payments, migrant remittances, credit, tax collection, and welfare benefits. These services depend entirely on the mobile phone network as their carrier, so they stop where the network does. This leaves millions of the very poorest people stranded – people living in remote areas where there is little to no network service. It also leaves urban users at the mercy of network congestion.

We developed a prototype system, DigiTally, which lets users make offline payments by copying short strings of digits from one mobile handset to another. Offline payments are already used for electricity (both in prepayment meters and pay-as-you-go solar); can we extend them into a general-purpose payment system, to increase service resilience in the face of network congestion or outage, and provide service to currently excluded areas?

We report the results of a preliminary study with an early prototype of DigiTally, tested on participants from a university in Nairobi (Kenya). The code-sharing process presented a possible usability challenge. To explore this and other aspects of an early prototype, DigiTally was introduced to Kenyan participants in order to resolve any major issues before a later field trial.

We discuss the lessons learned from our field visits and initial evaluation; we hope that this contribution is helpful for researchers and policy makers interested in mobile payments and financial inclusion. We also present our findings and observations. We found that, although offline payments involve copying codes in both directions between the payer's phone and the payee's, the extra workload was acceptable to most users.

Copyright is held by the author/owner. Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee.
Symposium on Usable Privacy and Security (SOUPS) 2017, July 12–14, 2017, Santa Clara, California.

1. INTRODUCTION

Mobile payments have transformed the lives of millions in less developed countries, bringing a means of exchange and a store of value to people who did not previously use a bank account or who lived far from any bank branch. They run on top of a mobile phone network; a user can typically load their phone with credit at the same agent where they buy airtime, and can send money to other service users. Over 200 such services have been launched worldwide and about 20 have achieved serious scale[1]; one pioneer was Kenya's M-Pesa [11], operated by the local phone company Safaricom. The initial killer application was migrant remittances, but M-Pesa is now very widely used for everyday purchases as well as specialist applications from paying pensions and government farm subsidies to collecting business taxes [12].

Because of the strong positive effect on development, the Bill & Melinda Gates Foundation called for innovations that could increase the uptake of mobile payments[2]. One of the largest impediments is that current systems operate entirely online; both the payer and the payee have to be able to communicate with the payment system server for a payment to be completed. This excludes millions of people living in remote areas with no network service; such people make up 10-40% of the population depending on the country. It also makes payments harder in the event of network congestion (we have observed 30-second delays in down-town Nairobi). Additionally, where the payment service operator is not the same firm as the mobile network operator, charges become an issue.

The main contribution of this paper is to describe a preliminary study that took place at Strathmore University in Nairobi, in September 2016. We set out to establish whether DigiTally was usable in three different environments: a coffee shop, a campus bookshop, and a cafeteria, and by students from a range of backgrounds. These students were experienced users of M-Pesa and thus able to compare it

[1]As shown on GSMA's 'Mobile for Development' website (using the *Mobile Money* filter): `http://www.gsma.com/mobilefordevelopment/tracker`

[2]Enable Universal Acceptance of Mobile Money Payments: `https://gcgh.grandchallenges.org/challenge/enable-universal-acceptance-mobile-money-payments-round-14`

with DigiTally; we also got assessments from the checkout staff. Further contributions are as follows: the paper analyses a security technology and offers lessons learned from a preliminary usability study. The paper should be of interest to researchers interested in development, and also to those interested in evaluating payment systems; both communities can benefit from our insights. Documenting the study in one publication should help them and others interested in this type of research.

We discuss the background and related work in section 2. We describe the technology in section 3, and we discuss our method and results in sections 4 and 5. We describe our observations in section 6, and provide more discussion in section 7. Finally, we present our conclusions in section 8.

2. BACKGROUND AND RELATED WORK

Modern mobile payment systems in less developed countries rely on encrypted Short Message Service (SMS) messages or Unstructured Supplementary Service Data (USSD) sessions, supported by a Subscriber Identity Module (SIM) card issued by a phone company. Examples include M-Pesa in Kenya [11] and bKash in Bangladesh [21, 5]. The SIM can verify a user PIN and contains keys for authenticating the customer; it can thus establish secure communication with a payment server. The server keeps everyone's transaction histories, just like in a conventional bank. Customers can check their balance and make payments using menu options on the phone's screen, displayed by the trusted SIM card. The phone number doubles as a bank account number. Cash-in and cash-out services are provided to customers by the mobile payment operator to facilitate exchanging fiat currency into electronic float and vice versa. These services are operated by a network of mobile money agents, who earn commissions based on the services provided (e.g. withdrawing funds from the system incurs a fee). To make a payment, a customer first enters the merchant's phone number (shops have numbers prominently displayed, with large shops having short codes). The customer must then enter the amount, followed by their PIN to authorise the transaction. A payment message is sent to the server; if the funds are available, they are transferred, and the merchant is informed. The merchant knows when the money has been received because their phone pings.

Jack and Suri discussed the economics of M-Pesa in [12], highlighting issues that affected system uptake, including liquidity and network reliability. Zimmerman and Baur discuss the challenges facing financial inclusion efforts [23], including network coverage and reliability, liquidity, complexity of user interfaces and payment processes, the lack of dispute resolution, and the lack of customer protection against fraud. Dupas *et al.* report that a significant proportion of the participants in their study listed fraud (embezzlements), unreliable services, and transaction fees as issues [9].

The goal of our work is to tackle the network coverage and unreliable services problems by processing payments *offline* reliably and deterministically, and to simplify the user interface as much as possible by mimicking familiar mobile payment systems. Furthermore, we aim to decrease transaction fees to encourage users to process transactions electronically. The purpose of this paper is to investigate possible usability challenges to be addressed in subsequent versions of our prototype. Offline payment systems have already been implemented, such as Geldkarte[3] in Germany and Net1/UEPS [1] in South Africa, but require dedicated devices or unfamiliar hardware that can be costly to replace if lost or stolen. Similar systems might perhaps be implemented on modern smartphones if both the payer and the payee had them. However, most users in less developed countries still use feature phones that do not have NFC, Bluetooth, or cameras; these phones cannot communicate data automatically in the absence of a network. We also minimise the assumptions we make regarding what features are available on users' phones.

Our goal is to design a system that operates without relying on such features, in case some of them are inoperable. For example, if we rely on a camera and it is damaged, the entire system is useless until the user has their camera fixed. By operating within these constraints, we are able to design a system that works on all low-end mobile devices; our project aims to provide a solution for the poorest demographics.

The use of feature phones emerged as a requirement during trips to rural Kenya. Users in Busia county, for example, specifically requested solutions that work on feature phones. One person commented: "*Don't give us [systems that work on] smartphones. We don't have those things and we don't know how to use them.*" When asked what kind of phones users owned, all but one person (in a group of more than 20), held up a feature phone; the non-feature phone was an old BlackBerry. Reliance on feature phones (called '*kabambes*' in Kenya) is due to both battery life and cost (in terms of price as well as maintenance). With such constraints, the only way to transfer information offline is for one phone to display it, and for a human to type it into the other phone (a similar approach used in device-pairing methods). Kainda *et al.* [13] looked at the tradeoff of usability and security with regards to different device-pairing methods, using out-of-band channels, as it applies to various device usage or capability restrictions. Their results show that typing strings (*"copy & enter"*) ranked first in terms of the aforementioned tradeoff. They also recommend that system designers take into account factors that affect that tradeoff including user conditioning, user motivation, security failures, and attentiveness. Moreover, transferring value by copying digits is well established in other applications, such as prepayment electricity meters [2].

3. TECHNOLOGY

An early design goal we set was that DigiTally must not require users to operate any unfamiliar hardware, and that the transaction flow should be as close as possible to the familiar one. This leads immediately to the challenge of programming feature phones: different operating systems make it difficult to implement and test applications for various models, leading to large costs for maintenance and support.

To minimise these costs within our constraints, the practical approach is to program the SIM card in the user's phone. SIMs were designed to host multiple applets in secure containers, which can prevent one applet from accessing other applets' data. A SIM provides a secure environment that we can control and that is compatible with all mobile devices adhering to the Global System for Mobile Communications (GSM) and smartcard interoperability standards. This

[3]Geldkarte website: `https://geldkarte.de/`

Figure 1: An overlay SIM (top) and a regular SIM (bottom). The top part of the overlay SIM can be peeled off and stuck on top of a regular SIM

means that we can program our system on a SIM, and insert it into any device that accepts a SIM. The user can move the SIM from one device to another, allowing for portability. SIMs also provide valuable built-in features, including atomic operations and rollback mechanisms, as well as the ability to store secure tokens and cryptographic keys in a tamper-resistant chip.

Feature phones normally have a single SIM slot already taken by the SIM issued by the Mobile Network Operator (MNO). MNOs do not generally let anyone else program their SIMs, but there is a new technology that can be used to circumvent this restriction. This is the overlay SIM (or sticker SIM): a SIM card only 120 microns thick that can be inserted between the regular SIM card and the phone. Figure 1 shows an overlay SIM and a regular SIM. Overlay SIMs were developed to support low-cost mobile roaming. They can also be used as a proof-of-concept prototyping environment, and to bypass MNOs' restrictions on devices if necessary. We used the programmable overlay SIM as a regular SIM, inserted in the single SIM slot. The same overlay SIM can also be stuck on top of an existing non-overlay SIM, and our system will work seamlessly on the device while allowing it to still work as a phone[4].

3.1 Overview of DigiTally

We designed and developed DigiTally as a Java card applet accessed through a user's phone (Figure 2). This applet can be loaded on an overlay SIM or a regular SIM. We chose to use a regular SIM for this study (for reasons we discuss in section 4.4 item 2). As discussed earlier, having an overlay-SIM ready applet enables deployment on feature phones even if the MNO chooses not to install DigiTally on their own regular SIM.

Rather than asking participants to use their own phones, we provided participants with dedicated phones (Nokia 130), preloaded with a SIM that included the DigiTally applet. The applet includes a menu that mimics existing mobile payment systems to capitalise on users' familiarity with them. The applet offers the user options such as `Balance`, `Send Money`, and `Receive Money` (Figure 3). Figures 8 and 9 illustrate the steps required to complete a transaction.

[4]Overlays SIMs are used in China for roaming purposes, and in Kenya by a local Bank to provide financial services to their users and to break local phone-payment monopolies. Overlay SIMs work on standard carriers' SIMs.

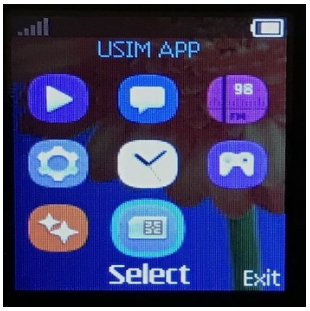

Figure 2: Selecting DigiTally applet from the feature phone's application menu

Figure 3: DigiTally user interface on a feature phone

The icon for the applet displayed in Figure 2 is enforced through the mobile OS. When a single SIM is present, selecting the SIM icon will display the applets that reside on that SIM. When multiple SIMs are present, such as in the case of using an overlay SIM on top of a regular SIM, selecting the SIM icon directs the user to two different menu options that represent the applet names in the two SIMs. Note that there could be more than one menu option displayed after selecting the SIM icon, if one or both of the SIMs offer more than one applet.

There are two main differences between traditional mobile payment systems and DigiTally. The first difference is that the SIM in an offline payment system stores the *balance* as a value counter, and the payment protocol changes this local balance in the SIM, whereas a traditional mobile payment system merely sends requests to a backend system to process operations on the user's balance. The second difference is the `Receive Money` option: in the absence of an online payment server, the recipient must be involved in completing a transaction. In traditional mobile payment systems, the recipient is not actively involved; they merely get an SMS saying how much money has arrived and from where. In offline payments, both phones must be involved in a transaction. The protocol is designed so that the parties learn immediately whether the transaction has failed or completed. Note that DigiTally does not require data to be sent over the network and can, therefore, work completely *offline*.

The technical description of the DigiTally protocol [4] describes the cryptography and other security mechanisms ([4] does not include a description of the design or implementation of DigiTally). For the purposes of this paper, we focus on the codes exchanged by the users to complete transac-

tions. These are Message Authentication Codes (MACs) that establish agreement between the payer and the payee on the transaction details, namely the payer, the payee, and the amount. These codes are computed using secret keys kept in their SIM cards based on the transaction inputs and previous history, and are designed to prevent transaction manipulation, replay, or guessing.

3.2 DigiTally Codes

We now outline the stages required to complete a DigiTally transaction. There are two codes involved in completing a transaction. To simplify the discussion, we will assume that Alice is paying Bob.

1. $Code_1$: After the payee B (Bob) has entered into his SIM the transaction amount X and the phone number of the payer A (Alice)[5], his SIM generates a random nonce (N_B), and then computes a MAC on A, B, N_B, X and the log of previous transactions ℓ between the two parties[6]. This MAC and the nonce together make up the 8-digit $Code_1$, which is shown on the payee's device (Figure 4). Alice similarly enters into her phone the amount X and Bob's number B[7]; it prompts her for $Code_1$, which she enters (Figure 5). If the two parties agree on X, ℓ and each others' identity, then Alice's phone accepts $Code_1$. If there's a disagreement – whether due to attempted cheating or an honest mistake – her phone will generate an error[8].

2. $Code_2$: If $Code_1$ is correct, Alice's SIM card decrements her account's available balance by the transaction amount X and generates $Code_2$ to authenticate the transaction. $Code_2$ is also 8 digits long; it consists of a 4-digit nonce N_A generated by Alice's SIM and 4 digits from a MAC on A, B, X, ℓ and N_B. Alice then shows or tells $Code_2$ to Bob (Figure 6). He enters $Code_2$ into his phone, and, if it is valid, his SIM increments his balance by X and a transaction log is displayed on his phone (Figure 7). A similar transaction log is shown to Alice to confirm the completion of the transaction (decrementing her balance by X).

This is the simplest payment protocol we could devise that enables value to be transferred from Alice's card to Bob's by copying 8 digits in one direction and 8 digits in the other. Its security is analysed and discussed in [4]. Here, our focus is usability; the STS prepayment electricity meters widely used in Kenya transfer value by means of a 20-digit number, presented as five groups of four digits [2]. A householder buys codes from an ATM or sales agent and copies them into their electricity meter; codes can also be bought online, using

[5]In our trial, user identities are randomly generated numbers, each simulating a phone number.
[6]If no previous transactions exist, then the first transaction initialises the relationship between the two parties.
[7]Alice and Bob can pick each others' names from a menu. The first transaction stores the contact's details, and subsequent transaction can later retrieve information from the locally saved contacts (on the phone or SIM).
[8]For example, if Alice enters \$4 on her device, and Bob enters \$5, then Bob's $Code_1$ will generate an error on Alice's device. The same thing happens if the wrong phone number is selected.

Figure 4: Bob's phone displays $Code_1$ that must be given to Alice

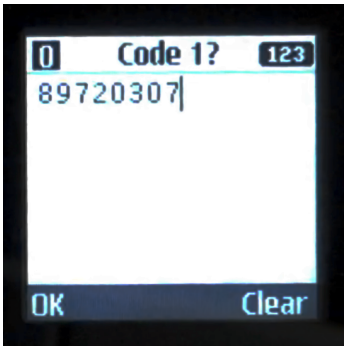

Figure 5: Alice enters $Code_1$ into her phone

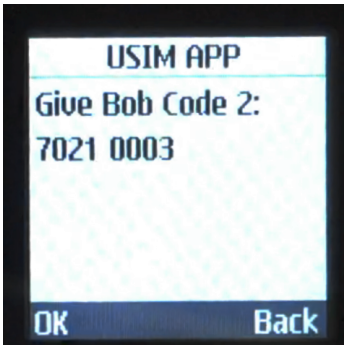

Figure 6: Alice's phone displays the response code ($Code_2$), given to Bob to authorise the payment

Figure 7: Bob's phone displays the transaction log after accepting $Code_2$

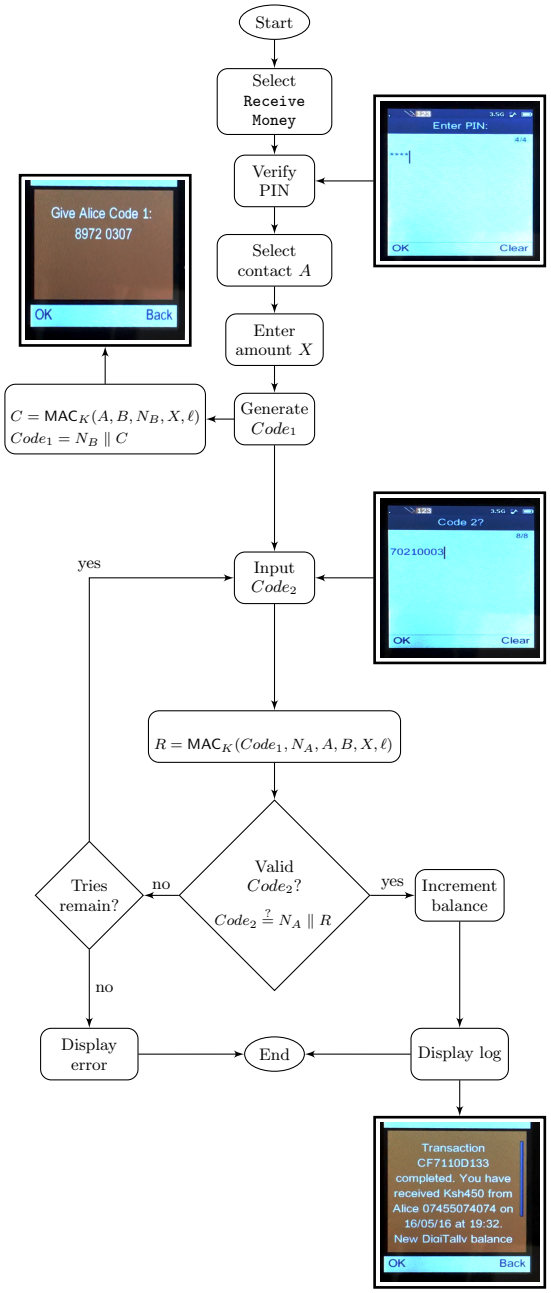

Figure 8: Recipient's (Bob B) transaction steps

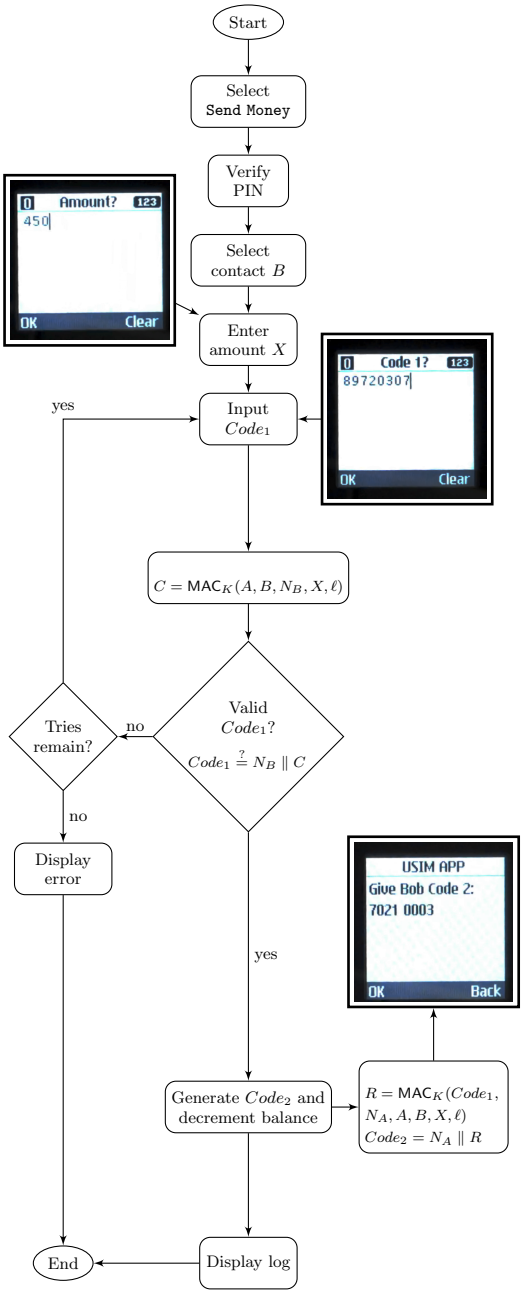

Figure 9: Payer's (Alice A) transaction steps

M-Pesa. Thus, we know that our prospective customers can copy digits, and that even illiterate people can use a phone. The object of the study is to establish whether the DigiTally transaction flow is usable.

Some care is necessary to ensure robustness in the face of mistakes. The DigiTally implementation includes various segments that execute atomic operations (enforced by the Java card platform) to ensure that critical components of the payment protocol are fully executed (or reverted to their initial state if a problem occurs). This provides the ability to reliably create *checkpoints*. We checkpoint the transaction when $Code_1$ is successfully generated so that a transaction can be recovered if, for example, a merchant accidentally generates a second, new $Code_1$ before the customer replies to the first one. Additionally, as Alice's balance is decremented prior to displaying $Code_2$ on the payer's device, we have to ensure that if she is interrupted (e.g. by a flat battery), her SIM can still retrieve $Code_2$ later ($Code_2$ is saved in the final checkpoint of the transaction, and can be retrieved from the "last transaction" log). Checkpointing was also tested (see section 6.2).

DigiTally codes do not have to be kept secret, since the codes can be used only once and only by the payer and payee in a specific transaction. There is no added risk if other

users observe the transaction or overhear the codes being exchanged verbally. This is in contrast to mobile payment systems that rely on secret codes: codes that compromise the users' security if intercepted, and increase the burden on the payer to deliver them to the recipient out-of-band. Users of feature phones can't use a secure messaging service, so probably have to use voice or SMS and hope for the best. As well as the (small) technical fraud risk, such codes can create anxiety and make dispute resolution problematic.

4. METHOD

The usability evaluation of DigiTally was conducted by the primary researcher from the University of Cambridge (UK) and researchers from Strathmore University (Kenya), in early September 2016. Participants from Strathmore University (hereafter referred to as 'the university') performed real-world transactions using DigiTally and then answered questions about their experience. As well as open ended-questions, we make use of the System Usability Scale (SUS), which is a standardised tool widely used for measuring usability. We give details below.

4.1 Participants

We wanted to test the technology as part of a pilot, and so we needed participants who could reveal weaknesses in our design before testing it as part of a field study with a representative sample. The aim was to maximise usability before introducing DigiTally in a larger-scale field test with target users. As an initial trial to establish any major issues with the DigiTally system itself, we recruited Kenyan participants from the university.

The participants were recruited following their registered interest in the study in response to an advertisement. Potential participants were interviewed with an aim to achieve diversity in terms of demographics, and to establish that participants would be able to give us detailed feedback to inform major re-designs before a field trial with a representative sample of rural users.

Twelve students and seven merchant staff were recruited as participants to use DigiTally for five days. Merchants were recruited from the university's cafeteria, bookshop, and a local coffee shop. According to Sauro [18], Vizri [22] and Nielsen and Landauer [16], very few participants are needed for early phase usability studies such as this, since adding more users tends to uncover the same issues, with 'five' often referenced as the "magic number" of participants. Confidence intervals may be wide as a result, but the average SUS score should be stable [17].

For wider coverage, we included a greater variety of participants, which required larger numbers than the proposed "magic number 5". This is in line with more cautious estimates of 10-20 participants for a usability test (e.g. Faulkner [10] and Macefield [15]).

In total, we had 19 participants for this preliminary study. There were 7 female and 5 male students, who were studying a range of topics in different faculties, including finance, law, and information technology. As for the merchants, the cafeteria included 1 female and 2 male staff members; the bookshop had 1 female and 2 males; and the local coffee shop had 2 females (only one of whom was responsible for processing DigiTally transactions). Each merchant was given one feature phone to process DigiTally payments; staff members in the cafeteria and bookshop shared the device to process payments, while the coffee shop had one person operating the feature phone.

Because we did not provide cash-in and cash-out services (to convert physical currency to DigiTally balances, and vice versa), participants were motivated to use DigiTally to spend the balance they had in their SIMs.

4.2 Evaluation Materials

Questionnaires and surveys are one of the most widely used methods for measuring attitudes [8]. They often involve asking participants to select one of a number of points on a rating (Likert) scale [20]. Good scales are valid (in that they represent the intended construct), and reliable (giving measurement consistency) [7, 20]. Standardised usability questionnaires and surveys are preferred because they are quantifiable, are economical to reuse, and allow for replicability of findings [20]. The most popular standardised survey for measuring attitudes towards usability is the SUS, favoured due to its brevity and being free to use [19]. The items in this survey factor into two sub-constructs of usability: system learnability (ease-of-use for new users), and system usability (defined as ease-of-use more generally) [19].

It was important to use feature phones for this study rather than the smartphones that many students normally use. Participants were thus given identical feature phones to test the basic usability of the technology first; a field test with representative users in rural Kenya who already use feature phones would be the next step.

We asked participants to complete a short pen and paper survey, which included the SUS, to make an initial assessment of DigiTally. Here, we asked participants for three free-text responses, to get insights into how they viewed DigiTally and where we might need to improve. These were:

1. What did you like about DigiTally?
2. What did you dislike about DigiTally?
3. Additional comments

4.3 Procedure

Students and on-campus merchants at the university were invited to try DigiTally and to give their opinions of their experience of the system. This involved using dedicated feature phones to transact using the DigiTally applet, which was installed on each phone's SIM. The students' phones were preloaded with a DigiTally balance of Ksh 2000 (about $19.50) that could be used to make transactions with merchant participants who accepted DigiTally. The merchants (cafeteria, bookshop, and a local coffee shop) were given phones that had a zero balance on their DigiTally applet.

Before the trial, participants gave informed written consent and were shown how DigiTally works. After the trial, participants were asked if they were willing to complete a survey about their experience, which was entirely voluntary. This survey consisted of the SUS and the three open-ended questions outlined in section 4.2 (Evaluation Materials).

Each day towards the end of business hours, the researchers visited each merchant to check their balances and reimburse

them in cash the amount they had in their DigiTally balance (after transferring the DigiTally balance to another phone, and thus resetting a merchant's balance to zero at the time of reimbursement). Each merchant signed a form indicating the date and amount of reimbursement. The reimbursement forms were necessary to create a detailed accounting of the money for our funders and auditors.

4.4 Ethics, Data Collection and Privacy

This study was approved by ethics committees at both universities involved in this research (one in the UK and the other in Kenya). To address the various concerns raised by the ethical review boards in both universities, we changed the design of the study to minimise any possible risks:

1. **Financial risks**: to address the risk of possible loss of personal funds, we provided the participants with a preloaded amount into their DigiTally applet (as explained in section 4.3). We decided not to provide cash-in and cash-out services to eliminate the possibility that participants would use their own funds to increase their DigiTally balances.
2. **Privacy risks**: we were asked for assurance that the overlay SIM would not compromise the privacy of participants by interfering with the participants' regular SIMs. Although we initially intended to use an overlay SIM on top of a regular SIM, we decided instead to use the overlay as a regular SIM and provide users with dedicated devices. The reason for this decision is that *proving* that an overlay SIM is not compromising users' privacy was out of the scope of this project. Note that the overlay SIM (and the DigiTally applet) does *not* compromise users' privacy in our implementation, since it operates as an independent sandboxed applet and does not interfere with other applets' data if there are any installed.

Therefore, we consider the preliminary study to have no significant ethical implications. There were no known risks to participants (financial, psychological, emotional, or physical). Participants were required to provide informed consent before starting the study. The informed consent form outlined what the study involved from the participant, how long it would take, our commitment to confidentiality, and their rights as a participant. Participants were reminded, verbally and in the consent form, that their participation was entirely voluntary, and that they may choose to not answer particular or all questions and may withdraw from the study at any point without having to provide a reason, and without fear of penalty from the researchers.

This study did not involve deliberately deceiving participants in any way. This study also did not involve recruiting participants from vulnerable groups, such as children, patients, people with learning disabilities, people engaged in illegal activities, or people in custody. Participants were fully debriefed and their questions were answered throughout the study. Additionally, participants were given contact information for all of the researchers from both universities should they have any further enquiries.

The data we used for our evaluation includes the survey results, as well as the performance and transaction data processed by the SIM. By collecting these data, we were able to analyse performance and usability issues as we discuss in Results (section 5). We collected the following:

1. Error rates (code-entry errors, and wrong PIN input)
2. Number of transactions
3. Number of attempts to unlock the SIM
4. Total amounts for all transactions (spent and received)
5. Transaction duration times

The SIM also stored the full log of the last transaction, which is overwritten when a new transaction is successfully processed[9]. This log contains $Code_2$ which can be retrieved after the transaction is completed. Recall that $Code_2$ can be used only once and with the right recipient (the identity of the recipient is one of the inputs required to generate the code), but a user might want to retrieve $Code_2$ in case their battery goes flat before exchanging the code with the recipient.

The collected data were stored in the SIM card (a trusted tamper-proof element). DigiTally does not leak any information outside this trusted element. Therefore, performance data are inaccessible without the correct authenticator. Only pre-configured devices, programmed to provide the correct authenticator, had access to the SIMs to obtain the performance measurements. Each SIM was programmed to lock itself after a certain number of failed attempts to access the contents of the trusted tamper-proof environment. At the completion of the preliminary study, we retrieved the phones from all the participants to extract the performance data from the SIMs and save it for later analysis. All the stored data were encrypted after extraction.

With the exception of participant names, no Personally Identifiable Information (PII) were collected in the study (such as date of birth, national ID numbers, etc.), as they were irrelevant to the scope of the study. We carefully explained to all participants what data were being collected, by demonstrating a full transaction on two phones and explaining the performance variables to be measured for the purposes of the preliminary study.

5. RESULTS

The following sections report on the actual error rates and transaction times, as well as the SUS results. We present descriptive statistics to help summarise the data and to show any emerging patterns. We make no claims about statistical significance, which would be reserved for inferential statistics. This would not be appropriate since the tested population do not represent the intended users of DigiTally, and were recruited in order to reveal any major usability issues with the technology itself. We thus make no claims about generalisability of the results.

We also discuss the open-ended results and the lessons learned through the feedback obtained from the users. We discuss

[9]This log is informational only, and is different from the cryptographic parameters securely stored on the SIM and used in the payment protocol as discussed in section 3.2. Refer to [4] for more details.

the features that participants considered to be the most important, and indicate issues that may need to be re-evaluated to improve usability.

Note that there is a difference between the total number of transactions completed by students, and the total number of transactions completed by merchants. This is due to the fact that students moved funds between their phones, perhaps to test the system and in some cases to demonstrate it to their friends; moving funds between devices back and forth did not incur any costs in terms of transaction fees.

5.1 Errors and Speed

Students completed an average of 23 transactions. The highest number of transactions completed by one participant was 30; the lowest was 8. For merchants, the average number of transactions was 59. The highest number of transactions completed by one merchant was 61; the lowest was 58. Tables 1 and 2 display a summary of the ***Error Rates*** and ***Time on Task***, which should be considered against the number of transactions each participant successfully completed.

5.1.1 Error Rates

DigiTally captured the number of errors participants made while trying to complete the task. Student participants made the most errors when entering the first code ($Code_1$) presented to them by the merchant. The number of errors here ranged from 0 to 9, with 7 of the 12 student participants making this error. These were non-critical errors since they do not prevent successful completion of the transaction. However, future trials will need to test the ease with which users can recover from this and other errors. For $Code_2$, the number of errors ranged from 0 to 8, with 5 of the 12 student participants making this error.

The merchant participants made the most errors when entering $Code_2$ which is required to authorise and complete the transaction. The number of errors here ranged from 4 to 6. Merchant $M2$'s $Code_2$ errors were higher because they included errors experienced during training. We adjusted the number in the table to reflect the errors experienced during the study without those experienced during the training. Merchants made no $Code_1$ errors; this type of error would occur if they used DigiTally to make payments instead of receiving (the merchants made a payment using DigiTally for the daily reimbursement transactions, see section 4.3).

We include PIN errors in Tables 1 and 2 for completeness. However, we do not consider entering PINs to be a great problem, at least no more than in any other system that uses them. For clarity, we report (in brackets) the proportion of errors relative to the number of transactions completed as a percentage (for non-zero values).

5.1.2 Time on Task

DigiTally recorded the time on task for each participant. The transaction timer starts when a menu option is selected (`Send Money` or `Receive Money`), and the timer is stopped before the transaction log is displayed to the user (the transaction log includes the transaction's duration time).

The average time for students to complete a transaction was 36.4 seconds. Their average completion times ranged from 24.2 seconds to 54.3 seconds. It is worth noting that a large number of errors by a participant did not necessarily translate into longer average time spent on transactions. For the merchant participants, the average time to complete a transaction was 51.23 seconds. Their average completion times ranged from 40 to 69.8 seconds. Merchants' transaction times are larger since a merchant can start a transaction, give $Code_1$ to the payer, then complete a few tasks to serve the customer until $Code_2$ is entered. Such tasks include having to prepare food at the same time as processing DigiTally payments, which is the case for merchant $M3$. For merchants dedicated to processing payments (cashier roles), the transaction times are lower ($M1$ and $M2$).

5.2 SUS Results

The SUS is not diagnostic: it will not reveal specific problems, but it does give an idea about overall *ease-of-use*, and whether significant changes might be needed.

To give a clear idea of the results, the SUS is calculated so as to provide a score out of 100. However, the SUS Score is not a percentage. A score of 68 actually falls at the 50th percentile (i.e. the average SUS score is 68). If the score is below 68, there are likely to be serious usability problems that need tackling. A score of 80.3 or higher is ideal.

The average SUS score for DigiTally was 78.8, which is considered 'Good', and would be given a 'B+' grade. The lowest score was 50, which is considered 'Poor', and is equivalent to a 'F' grade. This score was given by a merchant and was the only score considered 'Poor' by SUS standards. The highest score was 100, which is the best possible, and is equivalent to an 'A+' grade. Eight participants gave the equivalent of an 'A+' grade.

For merchant participants, the average SUS score was 71.4, which is considered 'Good' and equivalent to a 'C+' grade; for student participants, the average SUS score was 83.1, which falls just short of being considered 'Excellent', and is equivalent to an 'A' grade.

Although the SUS is intended to be a measure of ease-of-use, Lewis and Sauro argue that it can also be used as a measure of *learnability* (using items 4 and 10 of the SUS) [14]. As with calculating the SUS score, learnability can be calculated to give a score ranging from 0-100. The average learnability for this initial trial was 82.9, which falls just short of 'Excellent' and would be given an 'A' grade. The remaining 8 items are what Lewis and Sauro call a measure of *usability* [14]. For the current initial trial, this score was 77.8, which is also considered 'Good' and would be given a 'B+' grade.

5.3 Responses to Open-Ended Questions

Participants' answers to the three open-ended questions were categorised into several themes. This qualitative data analysis involved two researchers independently coding user comments to identify common themes, supported by quotations. These researchers then came together to assess agreement and categorised themes based on a well-established definition of usability (ISO-9241), which consists of effectiveness (usefulness), efficiency (ease-of-use), and satisfaction.

5.3.1 Perceived usefulness

1. **Money saving**. Most participants mentioned the benefit of there being no transaction fees. They pointed

	No. of transactions	PIN errors	$Code_1$ errors	$Code_2$ errors	Total code errors	Average time (seconds)
S1	30	2	0	8 (26.7%)	8 (26.7%)	30.9
S2	28	3	9 (32.1%)	0	9 (32.1%)	24.4
S3	18	3	2 (11.1%)	8 (44.4%)	10 (55.6%)	28.1
S4	22	0	9 (40.9%)	1 (4.6%)	10 (45.5%)	44.9
S5	29	1	1 (3.5%)	0	1 (3.5%)	24.2
S6	26	1	0	1 (3.9%)	1 (3.9%)	54.3
S7	26	0	1 (3.9%)	0	1 (3.9%)	50.9
S8	28	0	5 (17.9%)	0	5 (17.9%)	32.4
S9	10	0	0	4 (40.0%)	4 (40.0%)	28.8
S10	29	1	0	0	0	37.1
S11	8	2	0	0	0	42.1
S12	22	0	5 (22.7%)	0	5 (22.7%)	38.9

Table 1: The frequency and types of errors made by student participants and the percentage of each participant's transactions that their errors affected

	No. of transactions	PIN errors	$Code_1$ errors	$Code_2$ errors	Total code errors	Average time (seconds)
M1	58	2	0	4 (6.8%)	4 (6.8%)	40
M2	61	0	0	6 (9.8%)	6 (9.8%)	43.9
M3	58	2	0	4 (6.8%)	4 (6.8%)	69.8

Table 2: The frequency and types of errors made by merchant participants and the percentage of each participant's transactions that their errors affected

out that this would be useful for those in poor communities, as well as being attractive to price-sensitive customers and merchants. Aside from helping users avoid transaction fees, one participant mentioned that it also meant they did not have to use a smartphone, making DigiTally even more cost-friendly.

2. **Network independent transactions (interoperability)**. Many mentioned the benefit of not having to rely on network coverage. For some, this made the DigiTally transaction process seem more reliable. It also makes it more predictable as users do not have to wait for a confirmation SMS, which with M-Pesa can take up to 30 seconds.

3. **Security**. The general consensus was that DigiTally seemed very secure. The codes were a major factor behind this perception. While recognising this benefit, a few participants suggested that the codes were too long, and recommended that the developers consider shortening the codes.

4. **Money tracking**. Participants liked being able to review their last transaction and balance. For some, however, this was not enough. It was suggested that this feature would be more useful if the user could review all previous transactions.

5.3.2 *Perceived ease-of-use*

1. **Ease**. Participants perceived DigiTally to be simple and easy to use. Some clarified that DigiTally was easy to use *after* a learning period. One merchant described the process as cumbersome; this merchant was both serving customers and taking payments. Other merchant participants were either less busy or were dedicated cashiers. One participant also stated that DigiTally might be harder to use for the elderly and those with poor eyesight.

2. **Learnability**. DigiTally was most often praised for its learnability; in general, participants felt like DigiTally was easy for a first-time user to understand. Even the busy merchant ($M3$) who had complained that the system was cumbersome said she had no difficulty training a staff member to use it. The same merchant also warned that learnability might be lower for some, including the target population, where there might be more illiteracy and less education more generally. Another merchant was curious to know how money would be deposited in a production DigiTally system (this merchant is already an M-Pesa agent).

3. **Speed**. Participants found transacting with DigiTally to be relatively fast, once they knew the process. Two out of the eight merchants said that DigiTally had too many steps and was time-consuming compared to cash or M-Pesa. Requiring as much effort from the merchant as from the customer to give and receive codes was problematic when the merchant had a lot of customers and needed to perform other tasks at the same time. One merchant also did not like having to search for the customer in their contacts list. One participant pointed out that the speed of transaction would be less of an issue in rural areas.

4. **Errors and recoverability**. One participant stated that it is hard to make errors that would cause the user to lose money, and that they found errors easy to rectify. Many others did not agree. In general, error reset was considered too difficult. Although the process for error recovery was perceived as cumbersome, none of the users had to go through it during the trial. See section 6.2 for more details.

5. **Cashlessness**. Two participants noted that by using DigiTally they would not need to carry cash around. This was considered convenient. It was also noted that

merchants do not have to find the exact change when dealing with customers using DigiTally. This benefit is shared with other electronic payment systems; Kenyans already like M-Pesa as it dispenses with the risk and the inconvenience of cash.

6. **Codes**. Although many did not have much of a problem with the codes, finding them short enough and necessary for security, some stated that they did not like them, mostly because the code or the process was too long or awkward. Having to exchange and input codes was also perceived as an opportunity to make errors. On the other hand, one of DigiTally's perceived advantages was the deterministic nature of the transactions, because exchanging the codes provided immediate feedback that a transaction was completed, without needing to wait for an SMS.

7. **Distance**. Sending money to a remote payee was identified as a potential problem. It was pointed out that errors could be more likely and recovering from them would be harder if users were trying to complete a transaction at a distance.

5.3.3 Satisfaction

1. **Likeability**. Participants liked using DigiTally, using words like 'happy', 'good', 'best', 'seamless', 'enjoy', 'enthusiasm', 'encouraging', 'smart', and 'satisfied' to describe their experience. Comments suggest that likability could be improved by targeting issues associated with the length of the codes and with recovering from errors.

2. **Other services**. In addition to a merchant who wanted to know how to do cash-in and cash-out transactions, one student participant also indicated that they would like to be able to do this using DigiTally. We did not provide cash-in and cash-out services during the trial.

6. OBSERVATIONS

We describe in the following sections our observations during the preliminary study. These observations were documented during regular visits to merchants to answer their questions, and during daily visits to reimburse merchants for their DigiTally transactions (as discussed in section 4.3). We highlight two important observations, visual cues and the error-recovery process, and discuss the lessons learned.

6.1 Visual Cues

We observed that participants chose to display the codes so that the other participant could see them and enter them into their own device. None of the participants indicated that they exchanged the codes verbally, and some stressed that visual exchange is easier. In crowded areas, a verbal exchange of the codes could lead to misheard digits. The cafeteria is a crowded environment with long lines of customers, where a window separates the cashier and the customer and a small opening at the bottom of the window allows exchanging cash or passing cards (Figure 10). In this environment, we observed multiple DigiTally transactions where the only verbal exchange was to acknowledge that a transaction was completed by saying "*it's OK*", and in some cases just a nod. When we mentioned this observation, most participants agreed that "*it's much easier*", and gave the example of the cafeteria. In general, once participants had some experience with the system, we noticed that they would always show the code and never speak it even in quiet areas. In the transaction flow people developed, visual cues were used to complete each major step, starting with the customer declaring that they want to use DigiTally by showing a yellow label on the back of the phone that identifies the participant by name (Figure 11). However, we should not neglect the fact that some users in the target population for financial inclusion might be visually impaired.

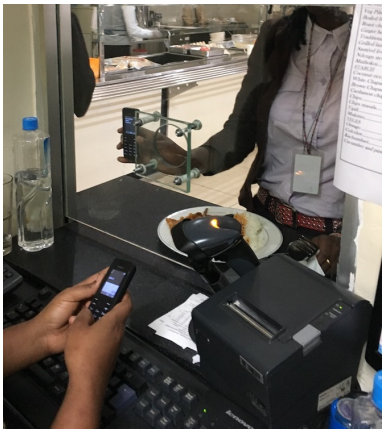

Figure 10: The cafeteria cashier (recipient) is operating the phone on the left, while a student participant (payer) is displaying $Code_2$ through the window to authorise the payment

Figure 11: Participants displaying the back of their phones (yellow DigiTally label shown on the back)

6.2 Error Recovery

We designed error recovery to prompt the user for three dummy inputs (any sequence of digits) to ensure that they were fully aware that they were indeed resetting the saved transaction data; in our implementation, a wrong reset could lead to inconsistent states on the SIMs involved. Inconsistent states currently require a 'hard reset' of the relationship between the two users by deleting the contact information and creating a new contact on each phone.

Before each prompt for $Code_2$, the recipient is shown a confirmation of the transaction details. After the dummy in-

puts are provided, the session is reset by discarding $Code_1$ as well as any intermediate results that were saved for validating $Code_2$. Now the recipient (merchant) can start a fresh transaction: the next time a transaction is initiated between the same two parties, a new $Code_1$ is generated. Even with default-text entry enabled in the applet to fill in the previously entered code, participants requested an easier way to rectify a mistake.

As error recovery was performed only by one student participant during the actual trial, most were likely evaluating the training they received rather than anything they actually did. Two users needed to reset a transaction during training. One participant commented: "*For the error issue: I think you should make it simpler to correct the error and provide instructions on how to correct it once the error is made since you will not be around to show them. For example, when an error of different amounts that needs one to put code 1 three times, I think you should write 'repeat code 1 three times to correct the error'.*" The other participant agreed: "*If there's a simpler way of resetting a wrongly transacted code DigiTally will be better.*" None of the other participants, including the merchants, had to use this error-recovery method, likely due to the training component instructing users to agree on the amount before proceeding with the transaction. We emphasised the importance of this step; perhaps the 'burden' of going through the error recovery process helped motivate careful exchange of the codes.

Our prototype error-recovery mechanism needs a redesign. A simple alternative would be a menu option to reset transaction data, requiring authentication with the user's PIN rather than requiring them to enter dummy inputs.

7. DISCUSSION

This paper reports the results of a preliminary study with an early prototype of DigiTally, tested on participants from a university in Nairobi. We described how DigiTally involves sharing two 8-digit codes to protect users from unauthorised payments while at the same time allowing them freedom from reliance on network coverage. This has an advantage that the payment service operator has no marginal transaction costs and can offer zero fees for some transactions. However, the code-sharing process has always presented a possible usability challenge. To test this and other aspects of this early prototype, DigiTally was introduced to Kenyan participants in order to identify and resolve any major usability issues before a later field trial with a more representative sample of service users.

The errors made in sharing codes suggest a need to make recovery from this type of error more intuitive. Nevertheless, most participants completed the task with few code-sharing errors. The average speed for every student was less than a minute, and overall the average transaction speed was close to half a minute. Given our observations of participants and the comments they made, it seems that the process, overall, seemed straightforward and fast. The SUS scale gives insight into the extent to which DigiTally's usability might inhibit its use. One of the lessons learned was the need to demonstrate clearly to users: (a) how errors can be avoided; and (b) how the codes prevent cheating.

Observations with initial trial users and open-ended answers following the trial indicate that users' main fear is the difficulty of recovery from errors. This is already an issue for M-Pesa when people send payments to the wrong phone number (it is a well-known problem, and one of the authors experienced it first-hand). Recoverability is especially important in the initial trials and adoption phase of a new system because expert help will not be available in remote villages; users' ability to figure out how to recover from mistakes may well be the difference between their adopting DigiTally and rejecting it.

Higher SUS scores tend to predict loyalty and word-of-mouth recommendations [18]. Users with scores over 80 (c.f. the average score of 83.1 for student participants) are called 'Promoters' because they are more likely to recommend a system, while users with a score below 60, called 'Detractors', are more likely to say negative things about a system. DigiTally is in a position to create 'Promoters', especially when it comes to student participants.

There are various things we can do to make DigiTally more usable for busy merchants. For example, we can give merchants a smartphone app that reads the customer's phone number from a QR code on a sticker on their phone, display $Code_1$, and read $Code_2$ from their phone screen. This way the merchant does no more work than with M-Pesa. Two of the merchants preferred DigiTally to M-Pesa because of speed; DigiTally does not force a cashier to wait for up to half a minute while a payment confirmation makes its way through Nairobi's congested mobile network. DigiTally takes more keystrokes, but the outcome is then immediate.

The fact that DigiTally has zero marginal costs also means that it can be offered as a zero-fee payment mechanism between friends and family, like personal cheques in the UK. Loan clubs, such as Rotating Savings and Credit Associations (ROSCAs) or savings clubs, and money-guards (used to enforce savings through the commitment of funds), whether formal or informal, are an important part of the financial ecosystem in many less developed countries. See Collins *et al.* [6] and Banerjee and Duflo [3] for more information about financial inclusion and the financial tools used by the poorest demographics (living on less than $2 a day).

Given our relatively high SUS, learnability, and usability scores, the challenge is encouraging first time use. The factor most apparently affecting first time use, based on free-text responses, was the 8-digit codes. The most frequent suggestion was to remove or reduce them. Although many participants were familiar with M-Pesa, and thus sharing an 11-digit phone number, this is potentially less time-consuming and error-prone because it is the same number every time for the recipient (their own phone number) and a single time entry for the sender. It also requires no input from the recipient, so a merchant can focus on other tasks.

Participants perceived DigiTally to be secure, and the codes were the reason behind this perception. As mentioned in section 5.3.1 item 3, participants asked if these codes can be shortened. However, this would not be possible without compromising the security of the system. We discuss the DigiTally protocol, the cryptography, and security parameters and features in the technical paper [4]. A few participants also requested a better money tracking tool: some participants liked that statistics about their transactions were available (e.g. amounts sent and received as explained in

section 4.4), and some users requested the ability to view more than just the last transaction (as discussed in section 5.3.1 item 4). This is not a critical issue, as we can engineer the system to include more transactions as allowed by hardware constraints and storage capacity.

Because migrant remittances are a key application of mobile payment systems (such as M-Pesa), some of our participants discussed the possibilities of remote transactions (see section 5.3.1 item 4). We designed DigiTally primarily for face-to-face offline transactions and not for remote transactions that would require a medium to exchange the codes. However, if such a medium exists, then DigiTally codes can be exchanged online to process remote transactions. Codes can be exchanged using SMS or over the phone; they could even be exchanged by post. DigiTally can be viewed as a platform to do the cryptographic operations required to process payments, on users' phones rather than on a centralised server, and systems can be built on top of DigiTally to perform remote transactions – relying on the DigiTally SIM to do the cryptography.

DigiTally was praised for being quick, easy to use, and easy to learn. Kenyans are already familiar with M-Pesa, which DigiTally was designed to mimic; that may have helped directly, while other factors our subjects liked, such as being able to review transactions, may have been features that they would like to see in M-Pesa. Therefore the results in this paper do not measure how DigiTally would be perceived by users who have never used any mobile payment system.

But M-Pesa, like most mobile payment systems, cannot work offline, and therefore fails to provide service to the poorest communities. Just as M-Pesa appealed most strongly to people with phones but no payment cards, so also DigiTally should appeal most strongly to people with no network service at all.

This was confirmed when we made two field trips to scout possible sites for a second-round field trial. When we visited a small town near Nairobi with good network coverage, stakeholders were interested in playing with the system, but saw its main benefit as being potentially programmable so that it could support their specific applications. When we visited Busia, a rural community near Lake Victoria with very poor network coverage, stakeholders were delighted, and played with it for hours. They considered DigiTally to be just what they needed to solve their problems with network coverage and reduce transaction costs.

8. CONCLUSION

We designed and developed an early prototype of DigiTally, an offline phone payment system, and tested it on Kenyan participants in a preliminary study. In addition to error rates, transaction speed data, and SUS scores, we reported on supporting data from demonstration sessions (observed behaviours and comments) and free-text written responses at the end of the study. Our results indicate that participants found DigiTally easy to use and that they liked key aspects of the system, including its perceived security and that it did not require network coverage to process payments.

We have demonstrated that DigiTally can be used for making payments without network coverage or transaction fees. Some specific technical improvements are needed, most notably the process of recovery from errors. We discovered that while DigiTally is slightly less convenient to use than existing mobile payment systems, the added burden is not excessive; people in areas with poor network coverage are eager to use it. Furthermore, the burden mostly falls on merchants, and busy merchants are likely to have enough money to buy better terminals. There is a realistic prospect of developing DigiTally into a workable system that will extend mobile payments to the millions of people who are currently excluded from the world of electronic payments.

9. ACKNOWLEDGEMENT

This work was supported by a grant from the Bill & Melinda Gates Foundation, and approved by ethics committees at the University of Cambridge and at Strathmore University.

We thank the following colleagues at the Computer Laboratory, University of Cambridge: Laurent Simon for providing feedback throughout the project; Alastair Beresford, Kat Krol and Alexander Vetterl for helpful comments about the paper. We also thank colleagues at Strathmore University: Edwin Njeru, Nelson Mutua and David Mutyethau for their assistance with this project.

10. REFERENCES

[1] R. J. Anderson. UEPS–a second generation electronic wallet. In *European Symposium on Research in Computer Security*, pages 411–418. Springer, 1992.

[2] R. J. Anderson and S. J. Bezuidenhout. Cryptographic credit control in pre-payment metering systems. In *Security and Privacy*, page 15. IEEE, 1995.

[3] A. Banerjee and E. Duflo. *Poor economics: A radical rethinking of the way to fight global poverty.* PublicAffairs, 2012.

[4] K. Baqer, J. Bezuidenhoudt, R. Anderson, and M. Kuhn. SMAPs: Short Message Authentication Protocols. *International Workshop on Security Protocols*, 2016.

[5] G. Chen and S. Rasmussen. bKash Bangladesh: A fast start for mobile financial services. 2014.

[6] D. Collins, J. Morduch, S. Rutherford, and O. Ruthven. *Portfolios of the poor: how the world's poor live on $2 a day.* Princeton University Press, 2009.

[7] J. DeCoster. Scale construction notes. *Department of Psychology University of Alabama*, 2000.

[8] R. G. Downey and C. V. King. Missing data in Likert ratings: A comparison of replacement methods. *The Journal of general psychology*, 125(2):175–191, 1998.

[9] P. Dupas, S. Green, A. Keats, and J. Robinson. Challenges in banking the rural poor: Evidence from Kenya's western province. Technical report, National Bureau of Economic Research, 2012.

[10] L. Faulkner. Beyond the five-user assumption: Benefits of increased sample sizes in usability testing. *Behavior Research Methods*, 35(3):379–383, 2003.

[11] N. Hughes and S. Lonie. M-Pesa: mobile money for the "unbanked" turning cellphones into 24-hour tellers in Kenya. *Innovations*, 2(1-2):63–81, 2007.

[12] W. Jack and T. Suri. Mobile money: The economics of M-PESA. Technical report, National Bureau of Economic Research, 2011.

[13] R. Kainda, I. Flechais, and A. Roscoe. Usability and security of out-of-band channels in secure device

pairing protocols. In *Proceedings of the 5th Symposium on Usable Privacy and Security*, page 11. ACM, 2009.

[14] J. R. Lewis and J. Sauro. The factor structure of the system usability scale. In *International Conference on Human Centered Design*, pages 94–103. Springer, 2009.

[15] R. Macefield. How to specify the participant group size for usability studies: a practitioner's guide. *Journal of Usability Studies*, 5(1):34–45, 2009.

[16] J. Nielsen and T. K. Landauer. A mathematical model of the finding of usability problems. In *Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems*, pages 206–213. ACM, 1993.

[17] J. Sauro. Measuring usability. *A Practical Guide to the System Usability Scale: Background, Benchmarks & Best Practices*, 2011.

[18] J. Sauro. How to measure learnability. `http://www.measuringu.com/blog/measure-learnability.php`, 2013. [Last accessed 01-November-2016].

[19] J. Sauro and J. R. Lewis. When designing usability questionnaires, does it hurt to be positive? In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2215–2224. ACM, 2011.

[20] J. Sauro and J. R. Lewis. *Quantifying the user experience: Practical statistics for user research*. Morgan Kaufmann, 2016.

[21] M. S. Uddin and A. Y. Akhi. E-wallet system for Bangladesh an electronic payment system. *International Journal of Modeling and Optimization*, 4(3):216, 2014.

[22] R. A. Virzi. Refining the test phase of usability evaluation: How many subjects is enough? *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 34(4):457–468, 1992.

[23] J. M. Zimmerman and S. Baur. Understanding how consumer risks in digital social payments can erode their financial inclusion potential. `https://www.cgap.org/sites/default/files/Brief-Understanding-How-Consumer%20Risks-in%20Digital-Social-Payments-March-2016.pdf`, 2016.