**System Requirements Specification Document (SRS)**

****

**Project 4 - E-commerce System for Marginalized Communities**

****



The purpose of this document is to define what the system must accomplish, according to the users’ requirements. During the analysis phase our primary focus is on what the users’ specifications are. How we deal with meeting these requirements is largely deferred to the design document. The requirements specification document provides detailed graphical models and narrative definition of the data and functional specifications.

**Team members:**

Ronald Chinku 10c4865

Giovanna Contu 10c1399

Dusan Gnjatic 10g0351

Ntsane Kolisang 97k5191

Tsungai Makoni 10m3716

Abram Rankapole 09r0881

**Supervised by:**

Ed De La Rey

Revision History

|  |  |  |
| --- | --- | --- |
| **Version and Date** | **Description** | **Author** |
| Version 0.1 (15 May 2013) | Initial Version: Theory | Abram, Ntsane, Ronald, Giovanna & Dusan |
| Version 0.2 (17 May 2013) | Second Version: UML diagrams | Tsungai & Abram |
| Version 0.3 (23 May 2013) | Third Version: finalization of document for iteration 1 | Abram, Ntsane, Ronald, Giovanna & Dusan |
| Version 1.0 (8 August 2013) | Fourth Version: Iteration 2 Use Cases, Context Diagram, Sequence and Activity diagram Change implementations | Abram, Tsungai & Giovanna |
| Version 1.0 (15 August 2013) | Fifth Version: Finalization of Iteration 2 requirements | Ntsane, Abram, Tsungai and Giovanna |
| Version 2.0 (3 September 2013) | Sixth Version: Iteration 2 modifications and Iteration 3 deliverables (All use case narratives, sequence diagrams, activity diagrams and state chart diagrams) | Giovanna and Tsungai |
| Version 3.0 (27 October 2013) | Seventh Version: Iteration 3 modifications as discussed with the group as well as proof reading and editing. | Giovanna and Tsungai |

Table of Contents

[1. Purpose of this document 6](#_Toc366768978)

[1.1. Scope 6](#_Toc366768979)

[1.1.1. Project Justification 6](#_Toc366768980)

[1.1.2. Deliverables 6](#_Toc366768981)

[1.1.3. Product-Related Deliverables 6](#_Toc366768982)

[1.2. Definitions, Acronyms 7](#_Toc366768983)

[1.2.1. Definitions 7](#_Toc366768984)

[1.2.2. Acronyms 8](#_Toc366768985)

[1.3. References 8](#_Toc366768986)

[1.4. Overview 8](#_Toc366768987)

[2. Overall Description of the Proposed System Product Perspective 9](#_Toc366768988)

[2.1. Product perspective 9](#_Toc366768989)

[2.2. Product Architecture 10](#_Toc366768990)

[2.3. Product Functionality/Features 10](#_Toc366768991)

[2.4. User Characteristics 11](#_Toc366768992)

[2.5. Constraints 11](#_Toc366768993)

[2.5.1. Interface requirements 11](#_Toc366768994)

[2.5.2. Performance requirements 11](#_Toc366768995)

[2.5.3. Security requirements 12](#_Toc366768996)

[2.5.4. Operational requirements 12](#_Toc366768997)

[2.5.5. Political and legal requirements 12](#_Toc366768998)

[2.6. Assumptions and Dependencies 12](#_Toc366768999)

[3. Specific Requirements 13](#_Toc366769000)

[3.1. Functional Requirements 13](#_Toc366769001)

[3.1.3. Actor Specification 15](#_Toc366769002)

[3.1.4. Use Case narratives 18](#_Toc366769003)

[3.1.5. Activity Diagrams 28](#_Toc366769004)

[3.1.6. Class Diagram 34](#_Toc366769005)

[3.1.7. Sequence Diagrams 35](#_Toc366769006)

[3.1.8. State Chart Diagrams 41](#_Toc366769007)

[3.2. External Interface Requirements 47](#_Toc366769008)

[3.3. Internal Interface Requirements 47](#_Toc366769009)

[3.4. Internal Data Requirements 47](#_Toc366769010)

[3.5. Design and Implementation Constraints 47](#_Toc366769011)

[3.5.1. Design Requirements 47](#_Toc366769012)

[3.5.2. Implementation Requirements 47](#_Toc366769013)

[3.6. Other Requirements or Constraints 47](#_Toc366769014)

[4. Operational Requirements 48](#_Toc366769015)

[4.1. Safety Requirements 48](#_Toc366769016)

[4.2. Security and Privacy Requirements 48](#_Toc366769017)

[4.3. Environmental Requirements 48](#_Toc366769018)

[4.4. Computer Resource Requirements 48](#_Toc366769019)

[4.4.1. Hardware Requirements 48](#_Toc366769020)

[4.4.2. Hardware Resource Utilization Requirements 48](#_Toc366769021)

[4.4.3. Software Requirements 48](#_Toc366769022)

[4.4.4. Communication Requirements 48](#_Toc366769023)

[4.5. Software Quality Factors 49](#_Toc366769024)

[4.6. Packaging Requirements 49](#_Toc366769025)

[5. Testing Provisions 49](#_Toc366769026)

[5.1. Visual Testing 49](#_Toc366769027)

[5.2. Functionality Testing 49](#_Toc366769028)

[5.3. Application and Integration Testing 49](#_Toc366769029)

[5.4. Network Connectivity Testing 49](#_Toc366769030)

[5.5. Acceptance Testing 49](#_Toc366769031)

[6. Table of Figures 50](#_Toc366769032)

[7. Table of Tables 50](#_Toc366769033)

**Table of Figures**

[Figure 1: Context Diagram 14](#_Toc370661886)

[Figure 2: Context Diagram 2 15](file:///C:\Documents%20and%20Settings\GIOVANNA\My%20Documents\Dropbox\SysDev_ItsAfrica%20Solutions\Analysis\FINAL%20SRS.docx#_Toc370661887)

[Figure 3: Use Case Diagram 16](#_Toc370661888)

[Figure 4: Activity Diagram (UC01 & UC02) 29](#_Toc370661889)

[Figure 5: Activity Diagram (UC03) 30](#_Toc370661890)

[Figure 6: Activity Diagram (UC04) 31](#_Toc370661891)

[Figure 7: Activity Diagram (UC05 & UC06) 32](#_Toc370661892)

[Figure 8: Activity Diagram (UC07) 33](#_Toc370661893)

[Figure 9: Activity Diagram (UC08) 33](#_Toc370661894)

[Figure 10: Activity Diagram (UC09) 34](#_Toc370661895)

[Figure 11: Activity Diagram (UC10) 34](#_Toc370661896)

[Figure 12: Class Diagram 35](#_Toc370661897)

[Figure 13: Sequence Diagram (UC01 & UC02) 36](#_Toc370661898)

[Figure 14: Sequence Diagram (UC03) 37](#_Toc370661899)

[Figure 15: Sequence Diagram (UC04) 37](#_Toc370661900)

[Figure 16: Sequence Diagram (UC05) 38](#_Toc370661901)

[Figure 17: Sequence Diagram (UC06) 39](#_Toc370661902)

[Figure 18: Sequence Diagram (UC07) 40](#_Toc370661903)

[Figure 19: Sequence Diagram (UC08) 40](#_Toc370661904)

[Figure 20: Sequence Diagram (UC09) 41](#_Toc370661905)

[Figure 21: Sequence Diagram (UC10) 41](#_Toc370661906)

[Figure 22: State Chart Diagram – Class: Product 42](#_Toc370661907)

[Figure 23: State Chart Diagram – Class: Product Picture 43](#_Toc370661908)

[Figure 24: State Chart Diagram – Class: Product Sale 44](#_Toc370661909)

[Figure 25: State Chart Diagram - Class: Sale 45](#_Toc370661910)

[Figure 26: State Chart Diagram – Class: Artist 46](#_Toc370661911)

[Figure 27: State Chart Diagram – Class: Community 47](#_Toc370661912)

**Table of Tables**

[Table 1: Acronyms 9](#_Toc370661931)

[Table 2: Actor Specification (AC01) 16](#_Toc370661932)

[Table 3: Actor Specification (AC02) 17](#_Toc370661933)

[Table 4: Actor Specification (AC03) 18](#_Toc370661934)

[Table 5: Use Case Narrative (UC01) 19](#_Toc370661935)

[Table 6: Use Case Narrative (UC02) 20](#_Toc370661936)

[Table 7: Use Case Narrative (UC03) 21](#_Toc370661937)

[Table 8: Use Case Narrative (UC04) 22](#_Toc370661938)

[Table 9: Use Case Narrative (UC05) 23](#_Toc370661939)

[Table 10: Use Case Narrative (UC06) 24](#_Toc370661940)

[Table 11: Use Case Narrative (UC07) 25](#_Toc370661941)

[Table 12: Use Case Narrative (UC08) 26](#_Toc370661942)

[Table 13: Use Case Narrative (UC09) 27](#_Toc370661943)

[Table 14: Use Case Narrative (UC10) 28](#_Toc370661944)

# Purpose of this document

The *Systems Requirements Specification* (SRS) document provides detailed graphical models and narrative definition (with a strong bias towards the graphical) of the data and functional specifications identified during analysis (the Requirements Determination and Requirements Specification phases). Essentially, this document precisely states the functions and capabilities of the E-Commerce for Marginalised Communities project where the aim is to develop a mobile application for the end user.

The expected audience of this document are the project team, ItsAfrica Solutions, the technical coordinator (administrator) of this project, and other stakeholders who will be involved in this project and the usage of the final deliverable.

## Scope

### Project Justification

The goal of this project is to provide a fully functioning system that will support the process of uploading pictures of arts and crafts to a web server called TeleWeaver in relation to an E-Commerce Mobile Application project for the Egazini Outreach Community Project to enable the end-user the ability to promote their artwork or crafts nationally and internationally. The system capabilities involved will include the ability to create, read, update and delete data via a mobile application interface. The product created should be future compliant in the event that other communities (other than Egazini) will also be involved in the Outreach Project.

* + 1. Deliverables

Project management-related deliverables: project charter, team contract, personality biases.

* + 1. Product-Related Deliverables
       1. Define the requirements of the system. This includes information about the main users of the system and the features of the system required to solve their business problems.
       2. The mobile application system should be tested continuously in order to discover bugs. Who will be doing the testing and how the system will be tested needs to be determined.
       3. Training will need to be conducted to the end user on how to use the mobile application system.
       4. A comprehensive help document will need to be developed for end-users of the mobile application system.

## Definitions, Acronyms

### Definitions

**Context diagram** – Puts the project into perspective by explaining where the system obtains its information, from which sources, and where the information is used. It shows how the system will consume inputs and export outputs to external entities involved.

**Class diagram** - defines the classes in the system by illustrating object-oriented programming features such as class relationships, operations and attributes.

**Wares** - artworks, crafts and any items wished to be sold on the e-commerce site.

**TeleWeaver** - a packaged “integrated enabler for rural TeleCentres and access nodes” (Reed House Systems 2011). It is adaptable and custom built to account for the limited resources of rural areas and works with a mobile middleware technology OSGI.

**Joint Application Development** - methodology that involves the client or end user in the design and development of an application, through a succession of collaborative workshops (Rouse, 2007).

**Unified Modelling Language (UML)** - the standard notation for describing and modelling object-oriented systems (Stumpf and Lavette, 2005).

**E-Commerce -** the industry that exists by conducting the buying and selling of products or services over an electronic system for example: the Internet or other online networks.

**Open Services Gateway Initiative -** “The Open Services Gateway Initiative is a consortium of more than 80 companies from around the world working together to create an infrastructure to enable the deployment of services over wide area networks to local networks and devices.” (Marples and Kriens 2001) It is a light, compliant technology that runs on Java-enabled phones. (Reed House Systems 2011).

### Acronyms

Table 1: Acronyms

|  |  |
| --- | --- |
| **UC** | Use Case |
| **E-Commerce** | Electronic Commerce. |
| **OSGI** | Open Services Gateway Initiative. |
| **UML** | Unified Modelling Language |
| **JAD** | Joint Application Development. |
| **AC** | Actors |

## References

MARPLES, D. and KRIENS, P., 2001. The Open Services Gateway Initiative: An Introductory Overview*. Communications Magazine, IEEE*, vol. 39, no. 12, pp. 110-114

Reed House Systems. (2011). Reed House Systems. Retrieved May 5, 2013, from Reed House Systems: <http://reedhousesystems.com/>

ROUSE, M., 2007. What is JAD? [Online]. <http://searchsoftwarequality.techtarget.com/definition/JAD>. [Accessed: 20th May 2013]

Stumpf, R.V. & Lavette, C.T. (2005) Object Oriented Systems Analysis and Design with UML, Pearson Prentice Hall.

## Overview

The aim of this project is to create an e-commerce mobile application system that will be used to upload or modify pictures, descriptions or prices of arts and crafts which the artists (end-users) choose to sell to the public. The mobile application is a small part of a bigger system that allows artists in a community the ability to place their goods into a portal for sale on the internet.

The mobile application will need to access the TeleWeaver server, in which the database for the project with all the wares will be situated. This database will be linked to an internal database on the mobile application.

The goal is to finish the entire system by the 25th of October 2013. The system must meet all written specifications, be thoroughly tested and be completed on time.

The system will be designed using the Agile Methodology technique, specifically the SCRUM technique and will be completed in four iterations. According to Cohn (2009), Scrum project management is defined as a software agile development process which allows projects to progress via a series of iterations called agile sprints. Furthermore, each sprint is typically two to four weeks long.

This document commenced after the JAD facilitated sessions/workshops were held with ITS-Africa Solutions, Alfredo Terzoli and the Egazini artists (who are the end users of the product), where the broad organizational requirements were established. The diagrams which are created for the system were produced using the UML technique. With reference to Stumpf and Lavette (2005: 422), UML is the standard notation for describing and modelling object-oriented systems.

This document is a living document and will be updated after each of the iterations to ensure it is up-to-date with any requirements that need to be modified, and to agree with any change requests that have been made throughout the process of creating this mobile application.

# Overall Description of the Proposed System Product Perspective

## Product perspective

The purpose of this project is to manage, develop, install, test and commission an android platform mobile application to be used by Egazini Outreach Project.

The purpose of the product is to enable Reed House Systems’ clients to use their phones to upload their artwork and crafts to the TeleWeaver database and thus be able sell their products to a broader market.

The product will serve as the front-end mobile application for the Reed House e-Commerce System. The application will connect to TeleWeaver using RESTful (REpresentation State Transfer) web services, through a Wifi connection. The mobile application will not have any control over anything on the administrator or TeleWeaver's side. Additionally it will not have any control over anything on the buyer's side. Finally, it will not have any control over delivery and payment systems. The product will however allow users to view summaries of their sales.

## Product Architecture

The product has five main components that encompass the system. They are:

* The User Access management process, which covers the login process and the user’s profile management (Users have to register with the Outreach Project face-to-face to obtain login details and set up a profile after which they will be able to manage their profiles by changing relevant information on their profiles).
* The Internal Database, which stores all the information of the user and the user’s products on the application’s own internal database before the synchronisation process occurs.
* Product management, in the form of adding and removing products, changing the picture (from mobile or by taking a new picture), as well as updating product details.
* Sales notifications and feedback, this takes the form of a sales report that serves to inform the user of any purchases of their artwork or crafts.
* The Synchronisation Process (Sending and receiving data to and from TeleWeaver), this encompasses the communication and synchronization aspect of the product and how the databases are able to exchange data and communicate with one another.

## Product Functionality/Features

The product serves as the front-end for an e-commerce system which is to be run as a mobile application. The essential functionality is that a registered user is able to login to the system with a username and password. Following which the user is able to upload a picture of his/her artwork or craft along with a price and description. The user will be able to modify any and all details including the picture of the artwork or craft. Each artwork or craft can have a maximum of four pictures uploaded.

Finally, the system will be able to communicate via Wifi and be able to send all data to the main TeleWeaver database. Additionally, the product will be able to receive information in the form of sales reports from the TeleWeaver database, again via a Wifi connection. The range of communication with the TeleWeaver database will be limited based on the range to TeleWeaver’s network.

## User Characteristics

The users have experience with mobile devices; however they will be inexperienced in terms of interfacing with software and interacting with systems via a Wifi connection. Hence, the system needs to be designed to cater for the fact that the users will not be very familiar with the use of such software. The system needs to be as user friendly and as simple as possible in order to allow users who have limited funds and access to Wifi to be able to conduct their operations as quickly as possible.

## Constraints

System constraints are described and explained under the following categories:

### Interface requirements

* Labels and headers need to be intuitive and understandable.
* Buttons and their functionality need to be clear.
* Must be suitably sized and scaled and hence readable for mobile devices.
* Needs to be usable with both a touch screen and non-touch screen mobile device.
* Text must be convertible and readable in different languages without impacting on the layout and display of the interface.

### Performance requirements

* System needs to be usable and run at a satisfactory level on any android mobile device
* Data sent to the external TeleWeaver database should take no longer than 3 minutes.
* Should take no longer than 3 minutes to receive any sales reports from TeleWeaver when connected to the network.
* Confirmation messages should be displayed within 0.5 seconds of completing an action.
* User should receive system feedback within 0.5 seconds.

### Security requirements

* User login information must not be stored on the mobile device apart from the username which can be ‘remembered’.
* In terms of data security and integrity, appropriate input must be entered, example, numbers for pricing of artwork and crafts.
* If a user forgets their login details, they will need to contact a representative from the Outreach Program to reset this password.
* The user will have no access to the database itself, and will only be able to communicate with the database.

### Operational requirements

* Usable only on devices with android OS software installed.
* Devices must be Java enabled.
* The communication between the system and TeleWeaver requires the user to have Wifi.
* The system requires permission to use the mobile device’s camera to take pictures
* The system also needs to have permission to store data in a folder on the device, for example storing pictures in the mobile’s gallery.

### Political and legal requirements

* The product must comply with all industry standards in terms of coding language.
* The product shall not use any terminology/linguistics that is exclusive to any particular area of South Africa.
* The product must comply with the policy on free and open software use for Android OS devices.
* For installation purposes, must request for the user’s consent to utilize the camera for the application as well as to store and manipulate data on the mobile device.

## Assumptions and Dependencies

* System must be usable by more than one user at a time.
* Requires a network connection of some kind (Wifi).

# Specific Requirements

## Functional Requirements

* + 1. **Context Diagram**

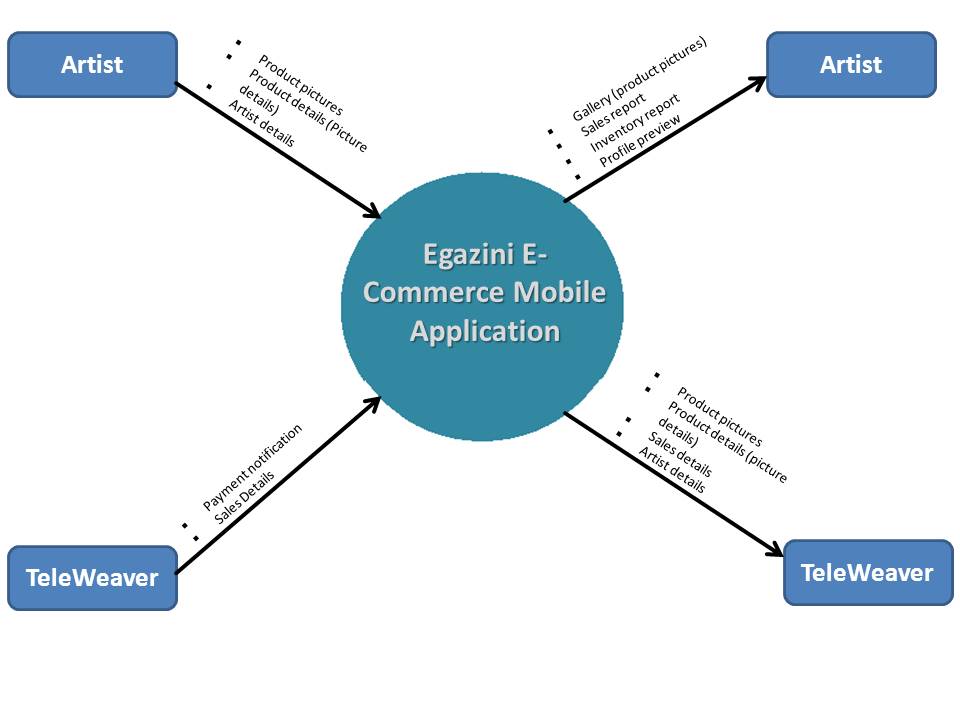
To fully explain the scope of the mobile application project we have provided two context diagrams. In the Figure 1 context diagram we portray the scope of the mobile application which an artist will use, which is what we intend to produce by the end of development. Artists are able to change details about themselves as well as their profile picture. Artists can also upload pictures of their products that they wish to sell and view, edit or delete these pictures as well as details about these products such as size or pricing. Artists will also be able to view their profile, their products and sales reports of items they have sold. These sales reports will be provided, through the TeleWeaver system to the mobile application, which can be accessed via Wifi services at any TeleCentre access node. The artist will have to register with the Outreach Project face-to-face where they will then be provided with login details to access the mobile application. The mobile application will have an internal database which can synchronise with the TeleWeaver database in range of a TeleCentre which will manage all necessary details.

Figure 1: Context Diagram

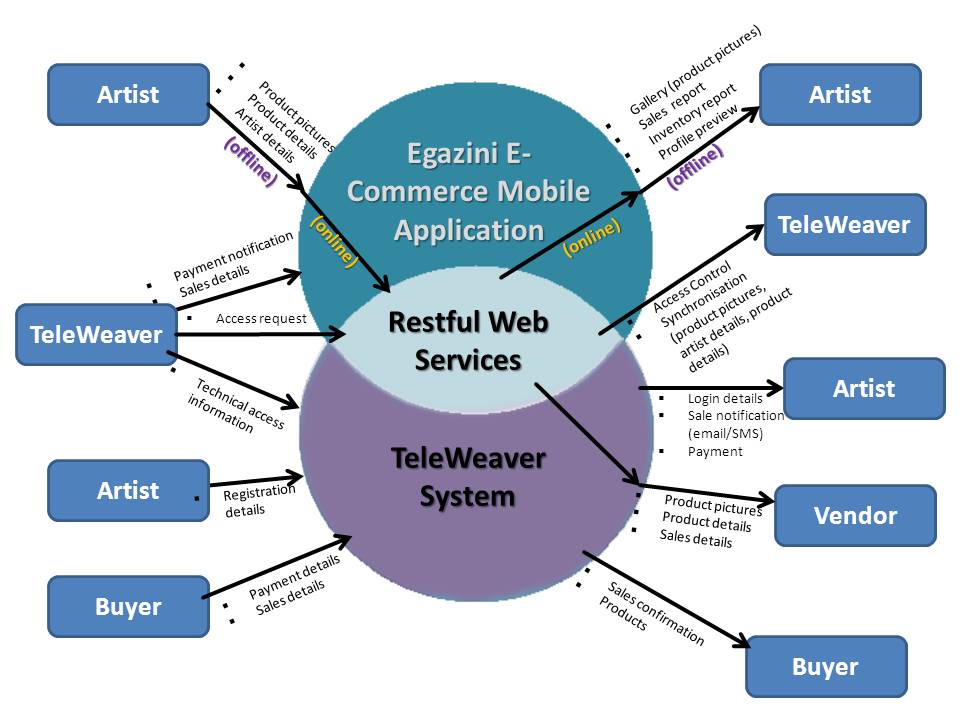
The Figure 2 context diagram is a different format of context diagram showing where this system will be placed in the broader spectrum of the project. As explained above, details will be able to be stored on an internal database until such a time as the application can synchronise with the TeleWeaver system. The application will connect to TeleWeaver using RESTful (REpresentation State Transfer) web services. In the broader spectrum of the project the mobile application does not interact with buyers or vendors of the sales items. All information from the buyers will be collected through vendor sites which have built-in payment systems. Vendor sites will connect to TeleWeaver through an abstraction layer and will have no connection to the mobile application. Once a sale has been made the artist will be notified via an SMS or email and not through the mobile application. However once the application has synchronised with the TeleWeaver database sales reports will be able to be viewed.

Figure : Context Diagram 2

* + 1. **Use Case Model**

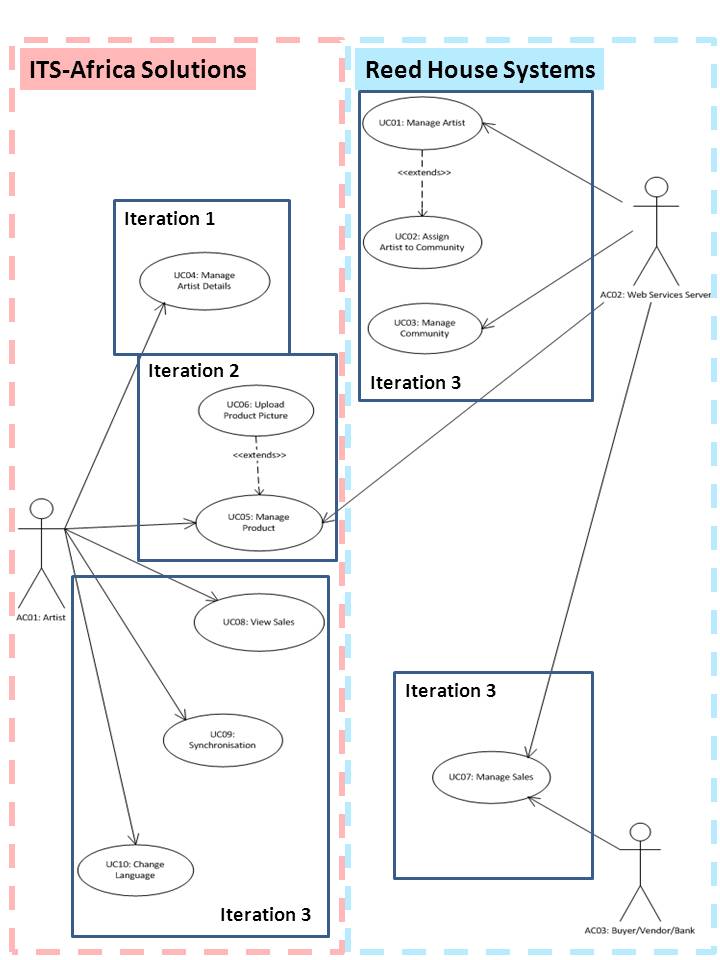


Figure 3: Use Case Diagram

### Actor Specification

Table 2: Actor Specification (AC01)

|  |  |
| --- | --- |
| **Actor Thumbnail** | **AC01: Artist** |
| **Actor Type & Stereotype** | This actor has administrative powers to change information related to themselves and their own artwork and crafts. |
| **Actor Description** | This actor has specific login details that allow them to access all information pertaining to their own artwork and crafts, which are being advertised for sale through the mobile application. This actor will also be able to modify any information pertaining to their own artwork or crafts, upload new artwork or crafts for selling, and view their sales reports. This actor also has the ability to change details on their own user profile (for example: profile picture or address). This actor will be responsible for linking their mobile application with the TeleWeaver server in order to ensure the correct information about themselves, and their artwork and crafts, is updated for sales purposes. Finally this actor will be able to change the display language of the mobile application. |
| **Actor Relationships** | UC04: Manage Artist Details  UC05: Manage Product  UC06: Upload Product Picture  UC08: View Sales  UC09: Synchronisation  UC09: Language Translation |
| **Interface Specifications** | None. |
| **Author:** | Tsungai Makoni |
| **Date Created:** | 17/05/2013 |
| **Last Modified by:** | Giovanna Contu |
| **Date Last Modified:** | 07/10/2013 |

Table 3: Actor Specification (AC02)

|  |  |
| --- | --- |
| **Actor Thumbnail** | **AC02: Web Services Server** |
| **Actor Type & Stereotype** | This actor has all the administrative rights in every aspect of the system. This is the actor who controls the artist’s administrative rights. |
| **Actor Description** | This actor essentially manages all information that is sent through the mobile application to the TeleWeaver server. This actor is able to view, register, update or delete new artists to the system as well as assign artists to communities. This actor provides the artist with login details to access an abstraction of the system. This actor will also be able to modify any information pertaining to the artwork of the artists on the TeleWeaver database. And, is responsible for managing information about the communities, adding new ones and deleting ones that are no longer involved in the Outreach Project. Finally, this actor is involved in the management of sales. Notifications need to be sent to the artists via SMS or email regarding the status of their products. This actor is also responsible for providing the product details and prices to vendor sites for sale on the internet and changing inventory attributes of the products, or removing items from the sales list when there are no longer any of that particular product left. |
| **Actor Relationships** | UC01: Manage Artist Details  UC02: Assign Artist to Community  UC03: Manage Community  UC04: Manage Sales  UC05: Manage Product  UC07: Manage Sales |
| **Interface Specifications** | None. |
| **Author:** | Tsungai Makoni |
| **Date Created:** | 17/05/2013 |
| **Last Modified by:** | Giovanna Contu |
| **Date Last Modified:** | 07/10/2013 |

Table 4: Actor Specification (AC03)

|  |  |
| --- | --- |
| **Actor Thumbnail** | **AC03: Buyer/Vendor/Bank** |
| **Actor Type & Stereotype** | This actor is only involved with the TeleWeaver system and does not have access to the mobile application. |
| **Actor Description** | Vendors gain access to the TeleWeaver database by requesting information about products that are ready to be sold and the quantities and prices of these products. (These products and information about these products are uploaded to the TeleWeaver Server via Restful Web Services when the mobile application has synchronised its data with the server.) Buyers can then make purchases of these products through these vendor sites such as eBay, Amazon and BidorBuy using the in-built payment system such as Visa, MasterCard or PayPal. |
| **Actor Relationships** | UC07: Manage Sales |
| **Interface Specifications** | None. |
| **Author:** | Giovanna Contu |
| **Date Created:** | 06/08/2013 |
| **Last Modified by:** | Giovanna Contu |
| **Date Last Modified:** | 16/10/2013 |

### Use Case narratives

Table 5: Use Case Narrative (UC01)

|  |  |
| --- | --- |
| **Use Case Specification** |  |
| **System Name** | **E-commerce for marginalised communities** |
| **Use Case Label** | UC01: Manage Artist |
| **Use Case Description** | Web Services Server is able to register new artists to the system, change information about these artists and delete information that is no longer necessary as well as delete artists that are no longer registered with the Outreach Project. |
| **Actors** | AC01: Artist  AC02: Web Services Server |
| **Pre-Conditions** | The artist must register face-to-face with the Outreach Project and provide their details which will be entered into the TeleWeaver system. The artist must have a mobile phone that is able to download the application and connect with the TeleWeaver database through Wifi. The artist also needs to be from a community that has registered with the Outreach Project. |
| **Post-Conditions** | Artist details will be saved on the system and the artist has been provided with login details to an abstraction of the system. |
| **Use Case Text (Basic Flow)** | 1. The artist registers face-to-face with the Outreach Project and TeleWeaver. 2. The artist’s information is entered into the system. 3. TeleWeaver provides the artist with login details to access the mobile application. 4. The artist can change or update any details as necessary. |
| **Alternative Flow** | 1. An artist that is no longer involved in the Outreach Project will have their login rights revoked. |
| **Author:** | Tsungai Makoni |
| **Date Created:** | 19/05/2013 |
| **Date last modified:** | 07/09/2013 |
| **Modified by:** | Giovanna Contu |

Table 6: Use Case Narrative (UC02)

|  |  |
| --- | --- |
| **Use Case Specification** |  |
| **System Name** | **E-commerce for marginalised communities** |
| **Use Case Label** | UC02: Assign Artist to Community |
| **Use Case Description** | Artists must belong to one of the registered communities in order to participate in the Outreach Project (e.g. Egazini). An artist can only belong to one community. |
| **Actors** | AC01: Artist  AC02: Web Services Server |
| **Pre-Conditions** | The artist must exist and the community must exist. The artist should live in a particular community that has been involved in the Outreach Project (e.g. Egazini). |
| **Post-Conditions** | The artist has been assigned to their community and is presented with login details that are specific to the region of the community. |
| **Use Case Text (Basic Flow)** | 1. An artist registers with the Outreach Project, and they belong to an existing community. 2. The artist is assigned to their community. |
| **Alternative Flow** | 1. An artist registers with the Outreach Project; however they do not belong to an existing community. 2. The project administrator decides if it is possible to add this artist to a new community. |
| **Author:** | Giovanna Contu |
| **Date Created:** | 08/08/2013 |
| **Date last modified:** | 27/10/2013 |
| **Modified by:** | Tsungai Makoni |

Table 7: Use Case Narrative (UC03)

|  |  |
| --- | --- |
| **Use Case Specification** |  |
| **System Name** | **E-commerce for marginalised communities** |
| **Use Case Label** | UC03: Manage Community |
| **Use Case Description** | The purpose of this use case is to register new communities with the Outreach Project. Information about these communities can be changed. Communities that are no longer registered with the Outreach Project can be deleted, provided that no artists are still assigned to that particular community. An artist can only belong to one community, but a community can have many artists. When an artist registers themselves with the Outreach Project their information will be recorded on the TeleWeaver database, which will provide them with login details. These login details will take them into their community’s site. |
| **Actors** | AC02: Web Services Server |
| **Pre-Conditions** | Communities must be willing and interested in registering with the Outreach Project, and the Outreach Project should have sufficient funds to account for the costs of adding a new community to the system. There should be artists in the community that are willing to sell their crafts on the internet with a communal billing service. |
| **Post-Conditions** | The community has been added to the Project and artists from that community can register and begin to make sales through the service. |
| **Use Case Text (Basic Flow)** | 1. Check for sufficient and willing community members to incorporate a new community. 2. Register the community’s name, location and logo with the Outreach Project. 3. Any changes to the community details can be made by the Teleweaver administrator. |
| **Alternative Flow** | N/A |
| **Author:** | Giovanna Contu |
| **Date Created:** | 08/08/2013 |
| **Date last modified:** | 27/10/2013 |
| **Modified by:** | Tsungai Makoni |

Table 8: Use Case Narrative (UC04)

|  |  |
| --- | --- |
| **Use Case Specification** |  |
| **System Name** | **E-commerce for marginalised communities** |
| **Use Case Label** | UC04: Manage Artist Details |
| **Use Case Description** | The purpose of this use case is to manage changes to existing artist details for a specific registered artist through modifying or updating. This can also be done by adding/revising information already on the system through the mobile application which will later be synchronised with the TeleWeaver database. |
| **Actors** | AC01: Artist |
| **Pre-conditions** | The artist must exist and must be registered with the Outreach Project and also have been assigned to a community. |
| **Post-Conditions** | The artist’s details will be updated and any changes made saved on the system. The artist can view the updated details. |
| **Use Case Text (Basic Flow)** | 1. The artist views their profile on the mobile application. 2. The artist identifies the information they would like to change. 3. The artist changes/updates the relevant information. 4. If satisfied with the changes made, the artist saves the updated information and can view it on the application. |
| **Alternative Flow** | 1. The artist may not be registered with the Outreach Project and thus may fail to login. |
| **Author:** | Tsungai Makoni |
| **Date Created:** | 19/05/2013 |
| **Date Last Modified:** | 08/08/2013 |
| **Modified by:** | Giovanna Contu |

Table 9: Use Case Narrative (UC05)

|  |  |
| --- | --- |
| **Use Case Specification** |  |
| **System Name** | **E-commerce for marginalised communities** |
| **Use Case Label** | UC05: Manage Product |
| **Use Case Description** | The purpose of this use case is to allow the artist to add, change or delete any artwork (as pictures with descriptions) to be for sale on the internet through the TeleWeaver system. Products also encompass quantity as an attribute, which the Web Services Server needs to manage when sales are made. |
| **Actors** | AC01: Artist  AC02: Web Services Server |
| **Pre-conditions** | The artist must exist, must have been registered with the Outreach Project and, successfully logged on to the mobile application. |
| **Post-Conditions** | A picture has been uploaded and can be viewed on the profile of the artist involved, which will eventually be synchronised with the TeleWeaver database. Any changes made are saved on the system. |
| **Use Case Text (Basic Flow)** | 1. The artist would like to add a new product for sale on his/her profile. 2. The artist identifies the product that they would like to add. 3. The picture can either be selected from their phone or a picture of the craft can be taken and uploaded to the application. 4. If satisfied with the changes made, the artist saves the picture and updated information about the product and can view it on the application. |
| **Alternative Flow** | 1. If not satisfied with the picture uploaded for the product (for example picture quality) the artist can cancel the picture upload or edit it later on the application. |
| **Author:** | Tsungai Makoni |
| **Date Created:** | 19/05/2013 |
| **Date Last Modified:** | 08/08/2013 |
| **Modified by:** | Giovanna Contu |

Table 10: Use Case Narrative (UC06)

|  |  |
| --- | --- |
| **Use Case Specification** |  |
| **System Name** | **E-commerce for marginalised communities** |
| **Use Case Label** | UC06: Upload Product Picture |
| **Use Case Description** | A product can only be added to the system by using a picture of that product (or up to 4 pictures of it) with a description to represent the product. This use case extends Manage product to incorporate a picture to represent the product. |
| **Actors** | AC01: Artist |
| **Pre-conditions** | The artist must exist, must have been registered with the Outreach Project and, successfully logged on to the mobile application. |
| **Post-Conditions** | A picture has been uploaded and can be viewed on the profile of the artist involved, which will eventually be synchronised with the TeleWeaver database. Any changes made are saved on the system. |
| **Use Case Text (Basic Flow)** | 1. The artist identifies a product that they want to add onto the mobile application for sale. 2. A picture of the item that needs to be sold is either uploaded from the phone’s memory or taken through the application at that instance. 3. The picture details are entered. 4. The product picture is saved onto the mobile application. 5. The picture can be changed as necessary. |
| **Alternative Flow** | 1. If not satisfied with the picture uploaded for the product (for example picture quality) the artist can edit the picture or cancel the picture upload. |
| **Author:** | Tsungai Makoni |
| **Date Created:** | 08/08/2013 |
| **Date Last Modified:** | 27/10/2013 |
| **Modified by:** | Giovanna Contu |

Table 11: Use Case Narrative (UC07)

|  |  |
| --- | --- |
| **Use Case Specification** |  |
| **System Name** | **E-commerce for marginalised communities** |
| **Use Case Label** | UC07: Manage Sales |
| **Use Case Description** | This use case encompasses information sent from the buyer through the vendor’s banking system to the web services server and then finally as a report to the artist through the mobile application. This use case allows the artist to view sales information about their products on the system. |
| **Actors** | AC01: Artist  AC02: Web Services Server  AC03: Buyer/Vendor/Bank |
| **Pre-conditions** | The artist must exist. The artist must have been registered with the Outreach Project, through TeleWeaver, and should be logged onto the system successfully. The artist must have a product uploaded onto the system with prices and sales information. |
| **Post-Conditions** | The artist is presented with the report that they have requested to view. |
| **Use Case Text (Basic Flow)** | 1. The artist views items that have been sold, or are pending a sale. 2. The artist can request to view specific Product reports on the mobile application. 3. A buyer will log onto a third party vendor website to make a purchase, vendor websites (e.g. eBay) will have access to the artist’s products through the TeleWeaver database. |
| **Alternative Flow** | 1. The artist may not be registered correctly with the Outreach project and then may fail to login and view sales. 2. If the synchronisation of the internal database of the mobile application with the TeleWeaver database has not been successful, products may not yet be for sale and therefore reports will not be generated. 3. The artist receives an SMS or email when a sale has been made, however the mobile application may or may not be able to immediately show this sale in a report depending on whether or not the artist is in range of a TeleCentre. |
| **Author:** | Giovanna Contu |
| **Date Created:** | 08/08/2013 |
| **Date Last Modified:** | 27/10/2013 |
| **Modified by:** | Giovanna Contu |

Table 12: Use Case Narrative (UC08)

|  |  |
| --- | --- |
| **Use Case Specification** |  |
| **System Name** | **E-commerce for marginalised communities** |
| **Use Case Label** | UC08: View Sales |
| **Use Case Description** | An artist is able to view all the sales they have made in the form of reports. Sales reports are important for the artists to keep track of how they are doing and make sure they are being paid the correct amount. |
| **Actors** | AC01: Artist |
| **Pre-conditions** | The artist must have products uploaded to the system, ready for sale, which have been synchronised with the TeleWeaver database. |
| **Post-Conditions** | The artist can review all the products he/she has sold, the quantity of that product sold, and the amount the product was sold for. |
| **Use Case Text (Basic Flow)** | 1. An artist views how their products are doing, and how many sales they have made. 2. The artist can also view specific reports about the Sales. |
| **Alternative Flow** | 1. The sales report might not be up-to-date if the mobile application cannot connect to one of the TeleCentres for synchronisation purposes. |
| **Author:** | Giovanna Contu |
| **Date Created:** | 08/08/2013 |
| **Date Last Modified:** | 27/10/2013 |
| **Modified by:** | Tsungai Makoni |

Table 13: Use Case Narrative (UC09)

|  |  |
| --- | --- |
| **Use Case Specification** |  |
| **System Name** | **E-commerce for marginalised communities** |
| **Use Case Label** | UC09: Synchronisation |
| **Use Case Description** | The purpose of this use case is to upload information from the mobile application’s internal database to the TeleWeaver database and vice versa. |
| **Actors** | AC01: Artist |
| **Pre-conditions** | The artist must exist, the artist must be in range of a TeleCentre, and the artist should have Wifi enabled on their phone. |
| **Post-Conditions** | The artist has updated all information on the mobile application database and on the TeleWeaver database. |
| **Use Case Text (Basic Flow)** | 1. An artist in range of a TeleCentre updates the information on the TeleWeaver database (e.g. Artist information, product information, new products) or on the mobile application database (e.g. Sales information, product quantity information) 2. The artists synchronises the data 3. Confirmation of successful synchronisation is given. |
| **Alternative Flow** | 1. The artist might not be in range of a TeleCentre or may lose signal in the process of the synchronisation process so the artist will have to try again when there is signal. |
| **Author:** | Giovanna Contu |
| **Date Created:** | 09/08/2013 |
| **Date Last Modified:** | 27/10/2013 |
| **Modified by:** | Tsungai Makoni |

Table 14: Use Case Narrative (UC10)

|  |  |
| --- | --- |
| **Use Case Specification** |  |
| **System Name** | **E-commerce for marginalised communities** |
| **Use Case Label** | UC10: Language Translation |
| **Use Case Description** | The purpose of this use case is to change the application’s display language. |
| **Actors** | AC01: Artist |
| **Pre-conditions** | The artist must exist and have successfully uploaded the mobile application onto their mobile phone. |
| **Post-Conditions** | The artist has successfully changed the application’s display language to another language in the repertoire. |
| **Use Case Text (Basic Flow)** | 1. The artist is given the option to change the display language of the application to one they prefer upon login. 2. A list of two possible language namely English of Xhosa is displayed. 3. The artists select the language they prefer. 4. All information is now displayed in the selected language. |
| **Alternative Flow** | N/A |
| **Author:** | Giovanna Contu |
| **Date Created:** | 03/09/2013 |
| **Date Last Modified:** | 27/10/2013 |
| **Modified by:** | Tsungai Makoni |

### Activity Diagrams

*UC01: Manage Artist extends UC02: Assign Artist to Community*



Figure 4: Activity Diagram (UC01 & UC02)

*UC03: Manage Community*



Figure 5: Activity Diagram (UC03)

*UC04: Manage Artist Details*



Figure 6: Activity Diagram (UC04)

*UC05: Manage Product extends UC06: Upload a picture*



Figure 7: Activity Diagram (UC05 & UC06)

*UC07: Manage Sales*



Figure 8: Activity Diagram (UC07)

*UC08: View Sales*



Figure 9: Activity Diagram (UC08)

*UC09: Synchronisation*



Figure 10: Activity Diagram (UC09)

*UC010: Language Translation*



Figure 11: Activity Diagram (UC10)

* + 1. Class Diagram



Figure 12: Class Diagram

### Sequence Diagrams

*UC01: Manage Artist extends UC02: Assign Artist to Community*



Figure 13: Sequence Diagram (UC01 & UC02)

*UC03: Manage Community*



Figure : Sequence Diagram (UC03)

*UC04: Manage Artist Details*



Figure : Sequence Diagram (UC04)

*UC05: Manage Product*



Figure : Sequence Diagram (UC05)

*UC06: Upload Product Picture*



Figure : Sequence Diagram (UC06)

*UC07: Manage Sales*



Figure : Sequence Diagram (UC07)

*UC08: View Sales*



Figure : Sequence Diagram (UC08)

*UC09: Synchronisation*



Figure : Sequence Diagram (UC09)

*UC10: Language Translation*



Figure : Sequence Diagram (UC10)

### State Chart Diagrams

*Class: Product*



Figure : State Chart Diagram – Class: Product

*Class: Product Picture*



Figure : State Chart Diagram – Class: Product Picture

*Class: Product Sale*



Figure 24: State Chart Diagram – Class: Product Sale

*Class: Sale*



Figure : State Chart Diagram - Class: Sale

*Class: Artist*



Figure 26: State Chart Diagram – Class: Artist

*Class: Community*



Figure 27: State Chart Diagram – Class: Community

## External Interface Requirements

The Basic4Andoid mobile application will have to store user related data in a database accessible through an Application Server called TeleWeaver. TeleWeaver will allow for the reception and delivery of services to and from the Basic4Android application. Thus the mobile application will need to interface with TeleWeaver in order for any data storage and retrieval.

## Internal Interface Requirements

The User Interface will be designed in such a way that it makes its usability as intuitive as possible. Its design will be centred around the core workflow process of the application. Furthermore users will not be able to personalize the screen layout in any way - layout changes will only be influenced by changes in data specific to each user.

## Internal Data Requirements

The E-commerce mobile application system requires: descriptions, prices, artists details, and the gallery of the artwork and crafts. The system must be able to store all the information on the server, which is in the back-end of the system when they are uploaded in the front-end.

The application tools that will be used are Basic4Android together with Microsoft Office Access 2010 to develop the system and develop the database diagram model of the whole system.

## Design and Implementation Constraints

### Design Requirements

Basic4Android is a new development platform that has been adopted by the department. Therefore, learning how to use the tool and where to find all the functionalities is challenging.

### Implementation Requirements

There are two implementers in the team which might not be well-aligned with the workload that will follow in all the iteration sessions until the conclusion of the system project.

## Other Requirements or Constraints

Other constraints that we foresee are time, as the time specified or given to complete the system is approximately a year. The requirements specification for the system commenced on 15 April 2013 and the system is expected to be completed by the 25th of October 2013. Time is a large constraining factor as the workload that can possibly be levelled out in years needs to be squashed all in a space of a year.

# Operational Requirements

## Safety Requirements

The terms and conditions of the application must be agreed upon and abided by.

## Security and Privacy Requirements

A protected password and unique username allow users to access the system.

Additional program functionality and features are restricted until authorization is granted.

## Environmental Requirements

The system needs to utilize as few resources as possible in order to conserve energy and processing power in order to reduce its impact on the environment.

## Computer Resource Requirements

### Hardware Requirements

Any mobile device capable of operating an android OS software.

### Hardware Resource Utilization Requirements

Utilize as few resources as possible process-wise, while being able to access and utilize the mobile device’s camera.

### Software Requirements

Be usable and compatible with android OS system.

### Communication Requirements

Be able to access and utilize the users’ mobile device’s Wifi to communicate with TeleWeaver.

## Software Quality Factors

The system needs to be operational and efficient from release with as few bugs as possible to ensure the system is usable and meets the standards and requirements set by the stakeholders.

## Packaging Requirements

The various operations must be grouped logically so the user knows where to find everything he/she needs.

# Testing Provisions

## Visual Testing

Usability testing commenced by using pencil drawn/stencil-based human machine interfaces being presented to potential users to interact with the screen so as to check if the interfaces are fulfilling their intended purpose. Next phase will encompass testing of the actual software human machine interface with potential users.

## Functionality Testing

Testing of the product’s functionality will entail the usage of the developer’s computer as a test bench to run the application and ensure that code is working as it should. The second testing that will be performed concurrently will be that of the database and this will be executed on the developer’s computer functioning as a test bench.

## Application and Integration Testing

The third testing that will occur will be that of integrating or connecting the actual application user interface with the database. Fourth testing consideration and provision is that of loading the complete application with the database on an android device and then running it to check for functionality and performance.

## Network Connectivity Testing

From testing the application on the android mobile device we will then test the network connection to TeleWeaver using the specified WebService protocol.

## Acceptance Testing

This testing will be done with the client and the end-user to ensure full compliance with the specified project scope.