

TEAM BRAINBOX

BOAFO

DESIGN DOCUMENT

MEMBERS

<i>CEO :</i>	<i>Michael Asare</i>
<i>VP Finance :</i>	<i>Debrah kwadwo Boafo</i>
<i>VP Marketing/People :</i>	<i>Anthonio Desmond</i>
<i>VP Engineering :</i>	<i>Ernest Pobee</i>
<i>VP Strategy :</i>	<i>Nyaku Selom Kobla</i>

20th july 2012

CONTENT

INDEX	TITLE	PAGE NO.
1.0	PURPOSE OF PROJECT	3
2.0	MODELS	3
	- ENTITY RELATION DIAGRAM	3
2.1	SERVICE MODEL	4
2.2	CATEGORY MODEL	4
2.3	SERVICE PROVIDER MODEL	4
2.4	SUBSCRIPTION MODEL	5
2.5	LOCATION MODEL	6
3.0	VIEWS	6
4.0	FLOW CHART OF CUSTOMERS	9
5.0	SERVICE PROVIDER TRANSACTION	10
6.0	USE SENARIO	11

1. PURPOSE OF PROJECT

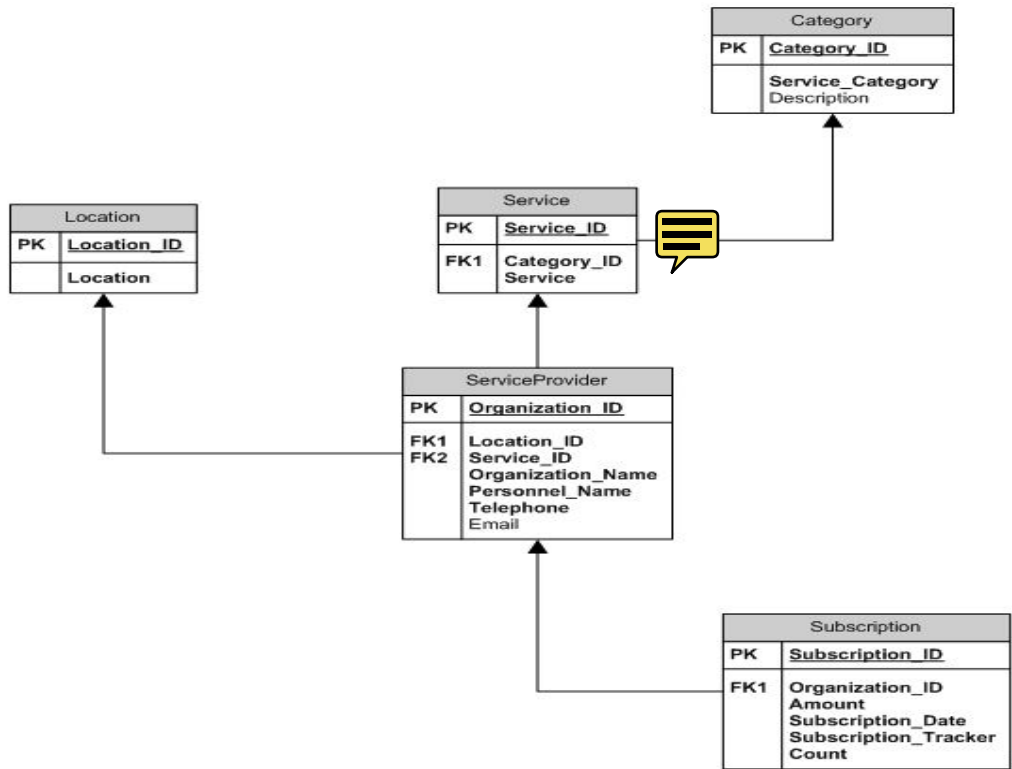
Boafo is a mobile sms application designed to connect its users to help. The main purpose of developing this application is to network people who are in need of a service to the people offering those services, this application is targeted at the general population in Ghana at one end and the service providers at the other end. Boafo targets Blue Collar jobs as its' service provider, giving the opportunity to that part of the population to find more customers and increase their earnings, this document provides a detailed design model to implement this idea.

The Boafo database is going to be developed using django and other means deemed appropriate in consultation with supervisors

2.0 MODELS

Boafo will have the following Models in its design, Service, subscription, category, service-provider and Location. Each model is elaborated below.

Entity Relationship Diagram



2.1 Service

This model will hold three fields Service_id, category_id, and service.

Service_id holds the the identification number of the service with which it can be referenced, it has a one to one relation because every service will have a unique service id.

category_id holds the identification number of the category and is a foreign key to the category model.

	Physical Name	Data Type	Req'd	PK	Notes
	Service_ID	LONG	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	This field uniquely identifies a servise
	Category_ID	LONG	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This is a foreign key to the Category table
	Service	TEXT(50)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This field refers to the name of a service
			<input type="checkbox"/>	<input type="checkbox"/>	

2.2 Category

This model will hold two fields, category and description.

category_id containing the identification number of the category as mentioned above.

Category will contain the category of the service and will have a one-to-many relation, linking one category to many service providers.

Description will have a one to one relation connecting each service provider to a brief job description.

Physical Name	Data Type	Req'd	PK	Notes
Category_ID	LONG	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	This field uniquely identifies a category
Service_Categ...	TEXT(50)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This field refers to the name of a category
▶ Description	TEXT(50)	<input type="checkbox"/>	<input type="checkbox"/>	This field provides a description about a category
		<input type="checkbox"/>	<input type="checkbox"/>	

2.3 Service Provider

This model will contain the following fields,organisation, name, phone_no, email, location, service, subscription_tracker,count.

Organisation field will contain the organisation of the service provider and will have a one to many relation, thus one organisation will have many services.

Name field will contain the name of the service provider and will have a one to one relation, that is every service provider has a unique name, the phone_no field and the email field will contain the phone number and the email address of the service provider respectively, and just like the name field will have a one to one relation.

Location field will contain the location of residence or work of the service provider and will have a many to many relation, being that a service provider could have many locations of work and many service providers can have one service.

service field will contain the service that the service provider is providing, this field will have a many to many relation as many providers can have one service and one provider can have many services.

subscription_tracker will keep track of the subscription status and when a subscription will expire.

count field takes count of the number of times the information of the subscriber has been delivered.

	Physical Name	Data Type	Req'd	PK	Notes
	Organization_ID	LONG	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	This field uniquely identifies an organization
	Location_ID	LONG	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This is a foreign key to the Location table
	Service_ID	LONG	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This is a foreign key to the Service table
	Organization_Name	TEXT(50)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This field refers to the name of the organization
	Personnel_Name	TEXT(50)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This field refers to the name of the personnel representing the organization
	Telephone	TEXT(50)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This field refers to the phone number on which the personnel can be reached
	► Email	TEXT(50)	<input type="checkbox"/>	<input type="checkbox"/>	This field refers to the email through which the personnel can be reached
			<input type="checkbox"/>	<input type="checkbox"/>	

2.4 Subscription

This model will contain the following fields, amount, organisation_id, subscription_id, count, subscription_date.

amount field will contain the amount the the service provider has subscribed, it has a one to many relation.

subscription_date field will contain the date at which the service provider subscribed.

	Physical Name	Data Type	Req'd	PK	Notes
	Subscription_ID	LONG	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	This field uniquely identifies a subscription
	Organization_ID	LONG	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This field is a foreign key to the ServiceProvider table
	Amount	LONGBINARY	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This field records the amount paid for a subscription
	Subscription_Date	DATETIME	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This field records the date a subscription was made
	Subscription_Tracker	BIT	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This field tracks the subscription status of a subscriber
►	Count	LONG	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This field holds the value of the number of requests a service provider's number has been dished
			<input type="checkbox"/>	<input type="checkbox"/>	

2.5 Location

This model contains the following fields, location_id, location.

location_id contains the identification number with which the location is referenced, and has a many to many relation.

location contains the location that will be referenced by the user and subscriber.

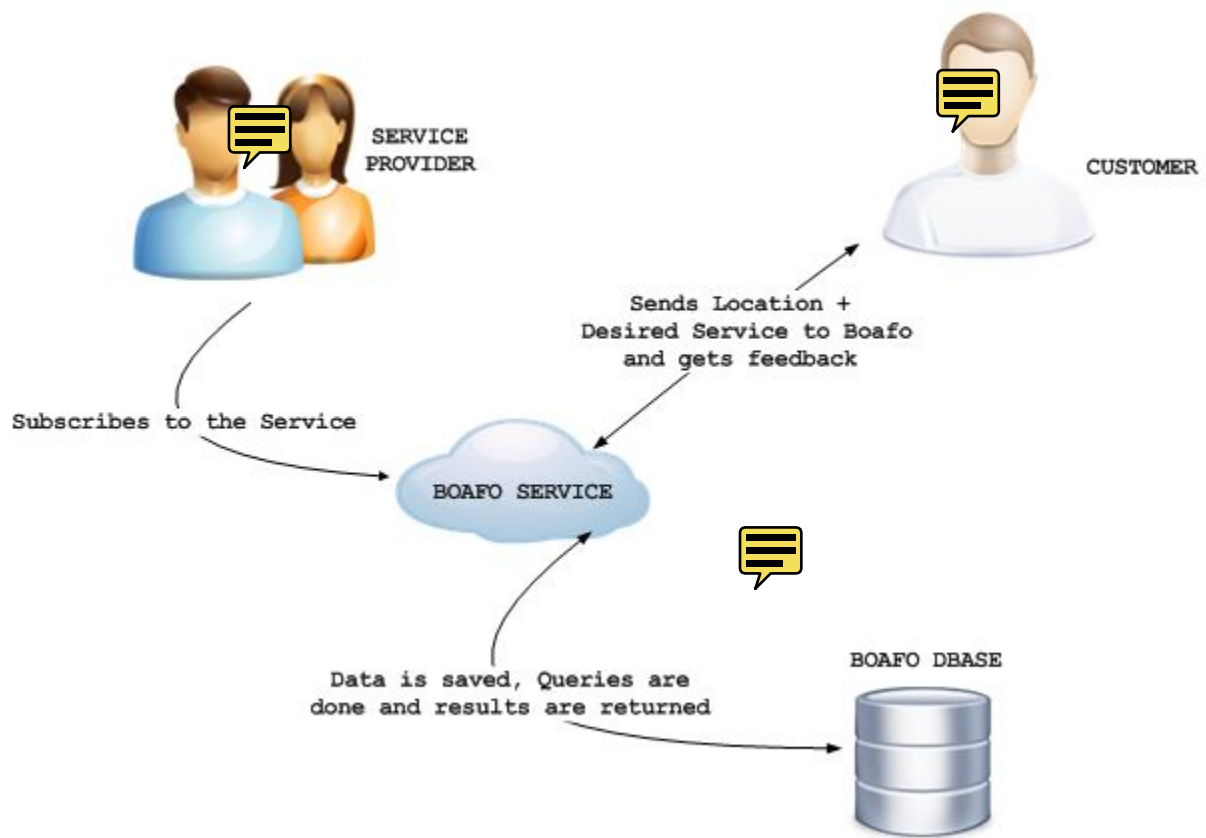
	Physical Name	Data Type	Req'd	PK	Notes
	Location_ID	LONG	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	This uniquely identifies a location
►	Location	TEXT(50)	<input checked="" type="checkbox"/>	<input type="checkbox"/>	This refers to the name of a location
			<input type="checkbox"/>	<input type="checkbox"/>	

3.0 VIEWS ASSOCIATED WITH THE BOAFO SERVICE

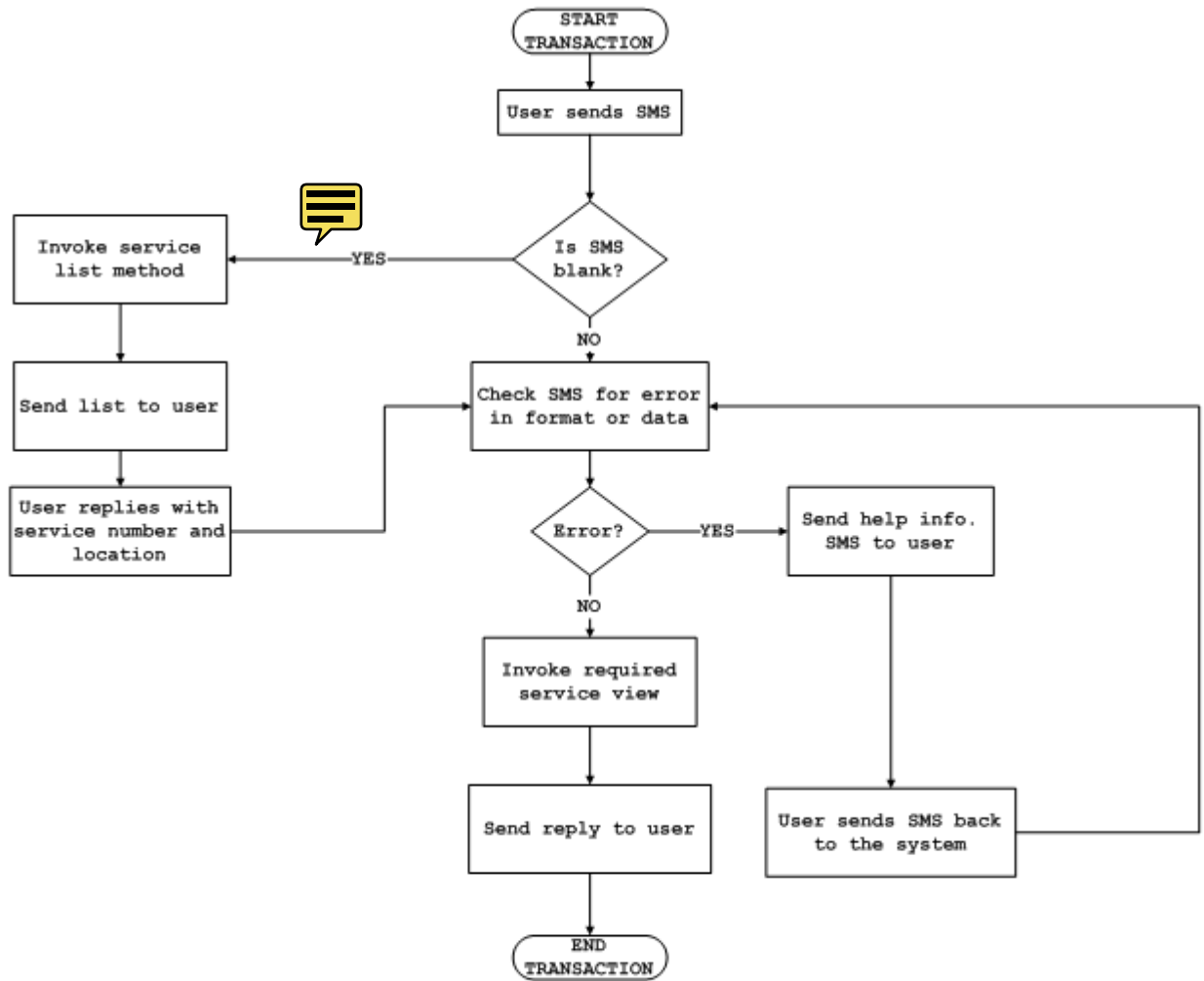
VIEW	DESCRIPTION
------	-------------

list_services(request)	This view just displays all services in the database
request_service(request)	This view handles requests from customers. If the SMS is blank it would invoke another view (list_service), if the SMS has the correct format , ie, 'service_no, location', then the view will invoke list_requested_service() to handle that request. if there is an error in the SMS format, the request_help() view will be invoked.
add_subscriber(request)	This view adds a service provider to our database.
add_service(request)	This view adds a service to our database
add_location(request)	This view adds a location to our database
edit_subscriber(request,id)	This view edits a subscriber based on the id provided
edit_location(request,id)	This view edits a location based on the id provided
edit_service(request,id)	This view edits a service based on the id provided
delete_subscriber(request,id)	This view deletes a subscriber based on the id provided
delete_location(request,id)	This view deletes a location based on the id provided

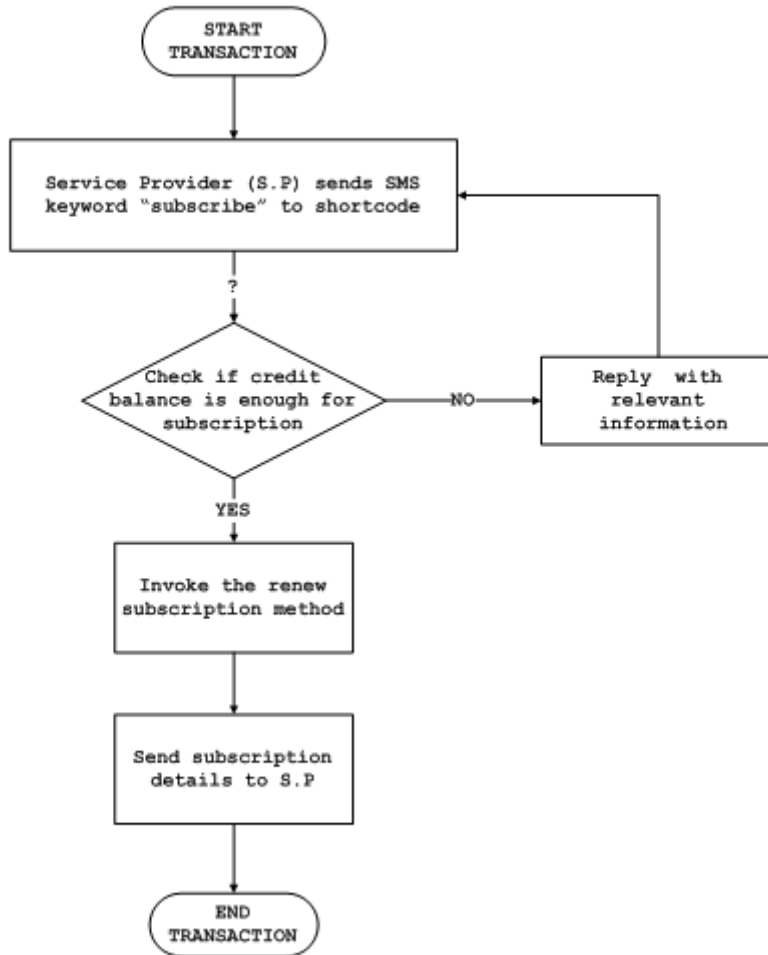
delete_service(request,id)	This view deletes a location based on the id provided
send_subscription_notification(request)	This view checks for all subscribers with outdated or pending subscriptions and send them either SMS or email notifications
check_subscription_status(request)	This view loops through all service providers in the database, checking their subscription date against the current system time. if there is a difference of 30 days, the subscription status of the service provider is set to disabled, else the subscription status is set to active for the month.
# rate_service_provider(request,id) #	



4.0 Flowchart of Customers



FLOWCHART FOR SERVICE PROVIDER TRANSACTION





Step 1: send empty msg to shortcode #####



Step 2: Boaf0 replies with list of services



Step 3: reply with category code and location



Step 4: Boaf0 replies with list of service

providers



Timeline for the Boafo project.

- By the end of this week we should be able to finish the business plan and executive summary.
- Business plan due is on 25th July, 2012. Hope to have developed our models and views.
- With the help and guidance we will receive from our instructors, Boafo service would be completed before the day of demonstration.