



the service providers(Safaricom, Airtel, Orange) computers.

If you didn't get the above I want you to think of any operation you do on your phone which is not sms, internet or making calls. Let's pick an operation, say checking credit or buying bundles (not bundles 'mwitu'). The process normally involves you entering a number in your phone with a unique structure.

***144# or *544#**

When you dial the above numbers you will get a prompt giving you either data or instructions to follow to get respective services.

This screen shows a sample request of user originated USSD request.

I believe by now you can relate to the technology. Now I want to jump straight and explain the magic that happens for this to be able to take place.

So basically what happens is this:

- User dials the provided USSD code.
- The request is forwarded to the mobile service provider.
- The provider routes the request through a gateway to the machine hosting the application.
- The application processes the requests it receives and sends back feedback.

- The feedback is processed by the provider and sent back to the mobile phone.

I am going to explain how to create a USSD application.

The magic behind USSD

For the spell to be successful you will need a magic wand which has several components:

- Postman Client for Chrome
- Running web server (LAMP, Xampp etc)
- Sublime text editor
- Brain
- Cup of coffee (several)

When you satisfy the above requirements, you will now be bestowed power by Dedi (the **name** of a fictional ancient Egyptian **magician**) to write and execute the spell.

Note that for this tutorial we are not going to use a live gateway but just simulate the way it would happen in the real environment.*

Spell 1

To develop for USSD there are key things you need to understand. Some of them are basic programming skills which include conditional statements, Switch statements etc.

The reason is because we are going to be doing a lot of string manipulation to extract data and interpret the request sent to us by the USSD gateway. One thing you should note is that the USSD gateways speak different languages i.e send strings with different naming.

For this tutorial we are going to adapt the following as our sample request parameters.

‘MSISDN’ => ‘254714611388’, (phone number sending the request)
‘SESSION_ID’ => ‘355562002’, (Session incase you need to perform audits)
‘USSD_STRING’ => ‘58*1234*456*252’, (This the string that contains the requests sent from the USSD gateway, basically it is the answers the user give when prompted on the phone.)

All the answers(data entered by user on mobile phone) are separated by ‘*’ and every answer the user enters is concatenated to the existing string. So you need to be conversant with explode() function for php.

Spell 2

This spell is now where the rubber meets the road. It involves creating a sample code do to emulate a process. For this course I will consider a registration system for people in a constituency.

One very important thing you need to know in USSD is the use of keyword CON and END.

CON keyword is used when implying to a continuous request. For example when you expect a user to enter an answer.

END is used to terminate the USSD session. When you append this at the beginning of you script, it will tell the provider that the session has ended and thus user is not given an input box afterwards but rather an 'ok' or 'exit'

Below is a sample php code which is also commented to make it easier understand what is happening.

```
<?php
```

```
#We obtain the data which is contained in the post url on our server.
```

```
$text=$_GET['USSD_STRING'];
```

```
$phonenummer=$_GET['MSISDN'];
```

```
$serviceCode=$_GET['serviceCode'];
```

```
#we explode the text using the separator '*' which will give us an array.
```

```
$level = explode("*", $text);
```

```
//check to see of the text variable has data to avoid errors.
```

```
if (isset($text)) {
```

The first request will have an empty text field and thus should show the welcome screen to the user. Note the user of CON keyword.

```
if ( $text == "" ) {
```

```

$response="CON Welcome to the registration portal.\nPlease enter you full
name";

}

```

We check each level to make sure it has data and has been passed or is next in line. Hopefully you will figure out what takes place here.

```

if(isset($level[0]) && $level[0]!=" " && !isset($level[1])){
$response="CON Hi ".$level[0].", enter your ward name";
}
else if(isset($level[1]) && $level[1]!=" " && !isset($level[2])){
$response="CON Please enter you national ID number\n";
}
else if(isset($level[2]) && $level[2]!=" " && !isset($level[3])){
//Save data to database
$data=array(
'phonenumber'=>$phonenumber,
'fullname' =>$level[1],
'electoral_ward' => $level[2],
'national_id'=>$level[3]
);
//Insert the values into the db SOMEWHERE HERE!!
We end the session using the keyword END.
$response="END Thank you ".$level[1].", for registering.\nWe will keep you
updated";
}
Also note that we echo the response for every successful request.
header('Content-type: text/plain');
echo $response;
}
?>

```

Spell 3

If you have endured to this point, I should give you a ninja sticker. :-)

The third spell will help us know whether our code is working or not. So to test the code download it from

<https://gist.github.com/drizzentic/18522c06e4529c58bc82fefe429e319b>

and follow this steps:

- Place the code in you webserver folder i.e htdocs or /var/www/html
- open postman client
- put the url to where the code is hosted e.g localhost
- add the parameters specified in spell 1 to the request url e.g

http://localhost:9000?MSISDN=0713038301&USSD_STRING=&serviceCode=*144#

- Kindly note that the USSD string should be empty for you to see the welcome page.
- send the request and check postman response window for the response from the server. e.g this represent the feedback the user gets on their mobile phone when the make a request.
- Modify the USSD string by adding you name e.g

http://localhost:9000?MSISDN=0713038301&USSD_STRING=John Doe&serviceCode=*144#

- Different prompt will come. To proceed to the next step put a '*' after your name to act as the separator. Do this for the rest of the prompts until you reach the end. e.g

http://localhost:9000?MSISDN=0713038301&USSD_STRING=johndoe*Nairobi&serviceCode=*144#

The above procedure will come in handy when you are developing USSD applications. Testing on the live environment is very costly, so before you can get a testbed to test, stick to postman.

Spell 4

It is not very important but you can as well do it. Add USSD as a skill to your linkedIn profile :-)