ESCUELA SUPERIOR POLITECNICA DEL LITORAL

EXPERIENCES BOOK

LENGUAJES DE PROGRAMACION

Computer Languages used:

- Java - Android
- Phyton
- Haskell

II PERIOD 2012-2013

TABLE OF CONTENTS

Índice

1. Title of the project: MaxSec App

1.1. Authors

* Mira Rodríguez Raúl Alberto
* Romero Triviño Jose Andrés
* Maya Herrera Ricardo David

1.2. Introduction

Of course it has spent to many of us that we are careless about our cell phone for a moment, and due to different factors somebody takes it without our permission; either some relative, or some partner of classes who wanted to play us a prank. Or on the contrary some of so popular ones friends of the foreign things could have taken our cell phone nowadays who take advantage of the first oversight some article about value avoids. Since our mobile application MaxSec was thought precisely for situation like previously named.

1.3. Functionalities

- Alert sound:

  MaxSec will count with one sound of alarm that will allow us to realize when our cell phone is taken without permission (using the sensor), This for cases in which the person that took our cell phone was remaining in a status of distance that was allowing us to listen to the above mentioned alarm sound.

- Location for GPS of the device:

  But there exist other situations in which for any reason either we do not manage to listen to the alarm sound, either be because there is a lot of noise in the place in which we are (example: a discotheque, a football stadium); or because the person that volume our cell phone moved away quickly from the place in which we are, or on the contrary we forget in some place our cell phone and to the moment that a person already takes it we are not close to it.

  For cases as these using the GPS of the cell phone, our application will allow us to know the exact coordinates in which our phone is in the moment in which this one is taken.

- It photographs in the moment of the alert:

  Apart from the functionalities already mentioned previously, MaxSec also will take a picture with the camera of the cell phone (also a picture was taken with the frontal camera in case it counts the device) at the moment when the alarm sounds, this for the case in which we could distinguish something from what appears in the photo, it is already some specific place or ideally some face.

- Notification by e-mail:

  After having compiled the earlier mentioned information (the coordinates as GPS, the photo), the user will be able to have access to it since this one will be sent by means of an e-mail to the indicated direction in the initial configuration of the application.

1.4. Detailed description

On having opened the application for the first time, this one will ask him to realize a small necessary initial configuration to be able to begin using the application. One of the most important information that will have to enter will be an e-mail address to which the notification mail was coming to him.
After having realized the initial configuration successfully the user will be able to visualize the principal screen of the application where he will have 3 options: RUN, OPTIONS y ABOUT.
RUN: In this option it will be possible to indicate if the application is .^{A}ctivated.^{o}r ”Deactivated”, in case of the application is activated it will begin working approximately 20 seconds after closing the application (depending on the configuration of the user).
OPTIONS: Here it will be possible to change the configuration of the application.
ABOUT: The information about the application and its developers.

- On having activated the application and having closed it, there will be a time of waiting of approximately 20 seconds that serves the user has time to leave the cell phone in a specific place where it is not going to move.

- Working the application this it will begin to sound already approximately 15 seconds after having felt movement (sensor), in this time the application will ask him to deposit a code to prevent the alert sound from being activated and from being sent the mail of notification, this in case the user has moved the cell phone for error.

- In case of the alert sound had been activated in this moment the local coordinates were taking in that the cell phone is and (s) photo (s) will be take it.

- After this the notification will be sent for mail to the mail specified in the initial configuration of application, in this mail the coordinates of the cell phone will be sent obtained previously, so also the captured photos.

1.5. General characteristics

The characteristics of the application are:

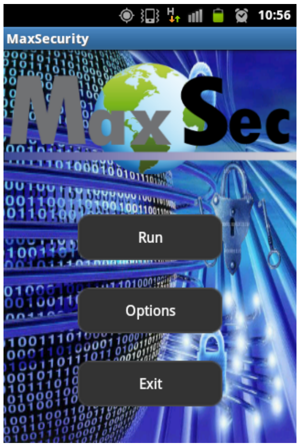
- Platform .- Android

- Programming Language .- Java.

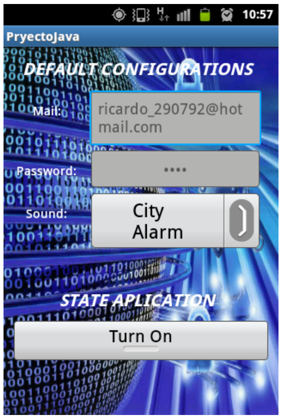
- Version No. 1
- Use of camera device Android
- GPS Global Positioning System
- Activation by means of the Sensor
- Programs to be used .- Eclipse IDE for Java developers
- Android SDK (android 2.1)
- Language Spanish

1.6. Manual Of Operation

1. We can observe the principal screen of our application with the buttons indicated in the image. In case of the GPS is not lit a message will appear so that the user lights it. After turning on we have to hope that the GPS should locate our position before pressing the button run.



2. On having pressed the Run button, this one throws another window with the mail for defect to which the application will order the mail, and its password so that no alarm is executed.



10:57

PryectoJava

DEFAULT CONFIGURATIONS

Mail: ricardo_290792@hotmail.com

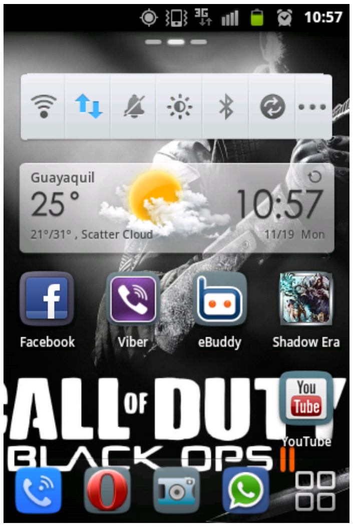
Password: ••••

Sound: City Alarm

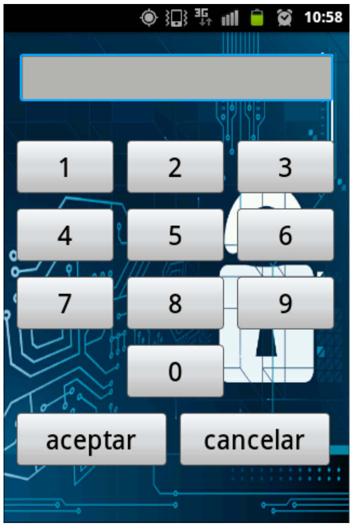
STATE APLICATION

Turn On

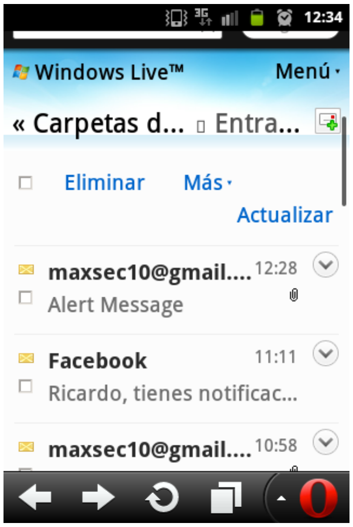
3. On having pressed the Turn ON button, the application is minimized waiting for a time period so that the user could make the device immobile.



4. On having felt some movement, the device throws a window of unfreezing which will be in charge of taking a picture and using the GPS to send to the wished mail (all this in the background), any time the code is incorrect the application he will take pictures and will remain blocked until the password is deposited correctly.



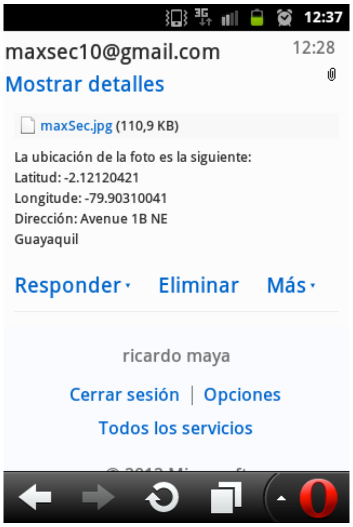
5.Here we can observe the mail that was sent to the formed e-mail, in which there appears the file of the image, the length, the latitude and the street in which the picture was taken.



maxsec10@gmail.... 12:28
Alert Message

Facebook 11:11
Ricardo, tienes notificac...

maxsec10@gmail.... 10:58



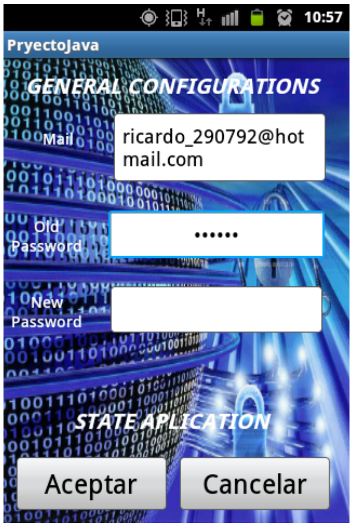
maxsec10@gmail.com 12:28

maxSec.jpg (110,9 KB)

La ubicación de la foto es la siguiente:
Latitud: -2.12120421
Longitude: -79.90310041
Dirección: Avenue 1B NE
Guayaquil

ricardo maya

6.On this screen it is possible to form which mail is wished by the user who sends to himself the image, also it can change the code any time the user knows which it was the previous one.



Mail: ricardo_290792@hotmail.com

Old Password: ••••••

New Password:

1.7. Starting using MaxSec

Before unloading and installing MaxSec, please to make sure that the application should support the type of phone. Once installed, to use for the first time the application, the mobile will must be connected to Internet (for the use of the GPS and to be able sending the corresponding e-mail of presenting the case to him).

1.8. Initial configuration

- Open the application and select the button Options to do an initial configuration
- In Options already fill the field Mail with your e-mail
- The field Old Password makes it empty to itself since as it is the first time that uses the application we do not have any previous password
- New Password there joins the password of unfreezing that you wish
- Finally press in the Accept button to confirm and to keep the configuration
- Otherwise press in the Cancel button to return to the principal screen

1.9. Block of the cellphone

After having carried the initial configuration out successfully we can light the application in such a way that the alarm is activated. For this we will have to continue the following steps:

- On the principal screen select the Run option
- Once there we will be able to see the e-mail that we deposit previously as well as also the sound of alarm that is selected
- To activate the alarm simply press in button Turn On

- After the application has done this it will keep on being executed in the background and the user will have a time of 15 seconds to leave the mobile in a place where it does not move.

1.10. Unfreezing of the Phone

Already blocked the phone, if someone moves it of his position automatically will show itself a screen where it was requested that the unfreezing password is deposited. After depositing the password and pressing in the Accept button the following stages might appear:

- If the password is correct the phone will have been unblocked successfully
- If the password is incorrect one will begin emitting the alarm sound
- If it presses in the Cancel button also the alarm sound will be emitted

1.11. Changing password or e-mail

In case the user should want to change his current password into a piece of news it will have to continue the following steps:

- Open the application and select the Options button.
- Once there fill the field Mail we replace the e-mail that is written by the new one that the user should wish, of the case was this, otherwise this field makes it equal to itself
- The field Old Password writes to itself the password that we are using at present and want to change
- The item New Password there joins the new password of unfreezing that you wish
- Finally press in the Accept button to confirm and to keep the configuration
- Otherwise press in the Cancel button to return to the principal screen

1.12. Changing Sound of the Alarm

The alarm sound will be able to change it at the moment in which it goes to activate the application that blocks the phone, on this screen we have the field Sound that will allow us to choose the sound of alarm of our preference

1.13. Remarks

Along the whole achievement of our project we have met certain difficulties that we want to mention in this document which we will mention them next:

- movement Sensor:

  MaxSec is provided with a sensor which allows us to know when someone makes use of our device after a certain moment. On having realized the code for the functionality of the sensor, android allow us to obtain the coordinates x, y, z of as the state of the device is in this moment. To be able to know when the movement takes place, what we do is to add these coordinates and the value to express it in absolute value it to be able to compare this way to a maximum number established by us, being so if the number added by the coordinates is major that the number established by us is because some movement took place. The problem takes root in that there are moments in which without moving the device it exceeds our maximum value. We believe that this sometimes happens due to the place where we leave our device, be already in some steep place or with relief.

- Apprehension of the Camera:

  At the moment of being able to capture the image after having deposited mistakenly the code, what we can visualize is a depth screen during a space of 2 seconds. This is the moment in which there is taken the picture, in which android allows us to realize it this way and not with the camera for defect, any time we create a window to be able to extract from which the image. So that the person does not know that this takes place in this moment, we did that this window is so small that it is almost invisible to the human eye.

- Use of GPS:

  At the moment of opening our application, it detects if the GPS of our device is active, in case is not, the length, latitude and direction from which the picture was taken will appear as null in the mail. Another disadvantage is the fact that, often on having identified our position, the object that provides us android to turn the parameters length and latitude in a direction does not recognize them, being so in the direction it sends us null.

- Notification by e-mail:

  The space that exists from the moment in which the picture is taken and the moment in which it is sent is 10 seconds. We did this this way so that no collision exists between the alarm and the moment in which the mail is sent, having time during these 10 seconds so that the alarm could dream without any problem.

1.14. Conclusions

We believe that our application is functional, knowing that there exist certain corrections and arrangements that should be done once let's know how to solve them, believing that they are not inconvenient for those persons who want to use it to be able to avoid these bad moments that often are our own friends those who make us happening.

1.15. Experiences of the development

Undoubtedly we are glad to be able could have known about one or another way this platform and this one without number of APIs that Android offers us. Many of us are used to using them but we believe that very few worry about as they work, now it is possible to say that we have a vision wider than it is the world of the programming and especially of as Android manages, and of course devoting ourselves understanding from that knowledge is needed on what it is the oriented objects programming, and that although very much code that serves to us for our applications is in the web, as nothing it serves if we have not the knowledge to be able to implement it according to our needs.

2. Title of the project: Lucas' Adventure

2.1. Authors

* Mira Rodríguez Raúl Alberto
* Romero Triviño Jose Andrés
* Maya Herrera Ricardo David

2.2. Introduction

Our Programming Languages' Project consisted on creating a game that could be played only with audio. It was not necessary to show anything on screen; the main objective of this project is that a blind person could play it. So we needed to develop this game in such a way that the user could put his earphones on and start this amazing experience that our story would offer him. This project didn't have to be a game necessarily, it could be also a story which the user could make decisions and depending on that the story will have several and different ends and outcomes. The direction of the story depends on the player, so he has to be careful with the decisions that he makes. So, we choose to create an adventure story about a young guy called Lucas who loves climbing mountains. This guy is so brave and intrepid so he decides to begin a new adventure. This adventure consists on climbing a mountain, but this won't be easy because this mountain has several obstacles and nobody has arrived to the top. Our project is based on Lucas' story and his risky journey to the top of the mountain, the user will be able to decide which way Lucas should take, and depending on that Lucas could reach or not his goal

2.3. Objectives

- To learn how to use the programming language Python.
- To learn how to develop application using Python.
- To compare the language Python with others with those that we have worked before.
- To understand the syntax of Python.
- To develop an application using the knowledge acquired about Python.

2.4. Project's Description

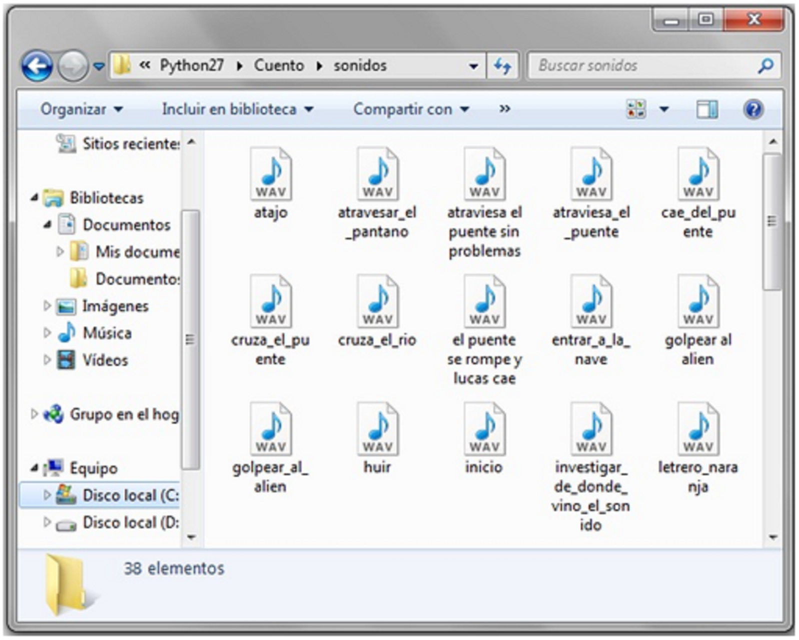
The project is an application, in our case it is a story where the user can make decision about the directions or the ways that the main character should take to reach his goal. This story was created by us and it is about a brave young guy called Lucas that wants to climb a mountain, and this mountain has several obstacles so it won't be easy for Lucas. This application was developed using the programming language Python 2.7.

For the using of sound we had to use a specific library called Pygame that is more focused in the development of games.

It is an application that can be used by any people, and this includes people that can't see because this game works only with to keys, the only thing that the user has to do is listen and press the button 1 or 2 to make his decision. As I said before, this is a game that also python™

pygame

can be played by blind people, so we decide to focus more in the part of the Audio instead the part of the Graphics. We didn't need to do develop a very graphic application so our project is only a window with a black background, because the really important thing about the project is sounds folder.



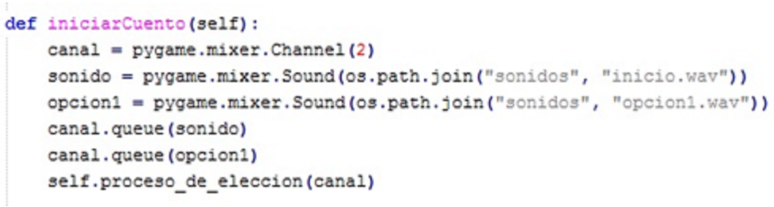
atajo
atravesar_el_pantano
atraviesa el puente sin problemas
atraviesa_el_puente
cae_del_puente
cruza_el_puente
cruza_el_rio
el puente se rompe y lucas cae
entrar_a_la_nave
golpear al alien
golpear_al_alien
huir
inicio
investigar_de_donde_vino_el_sonido
letrero_naranja

38 elementos

2.5. Implementation

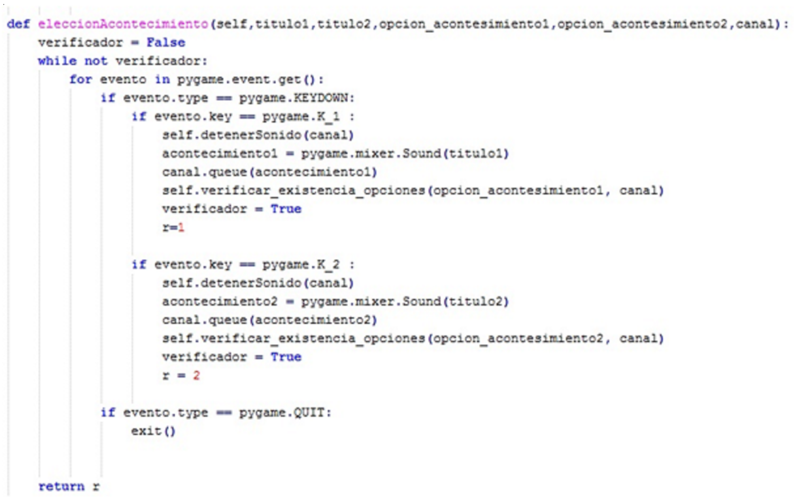
To develop this application we needed to learn how to program using a new programming language (Python) from zero, so we had to learn about the syntax of this languages and how to define functions and all this kind of things. We learned the basic about Object-Oriented Programming in Python so for our project we create a class called "Principal" and inside this class we have about 10 functions.

This wasn't a too extensive project so we didn't need too many functions; actually our class Principal isn't too extensive too. Besides using Python you don't need to write too much code to do something. Now we are going to talk about some functions very important for our application, and one of those is "iniciarCuento", because this is the one that starts everything, this function allows us to listen the beginning of the story and after that it allows to listen the two possible options that Lucas could choose. How to choose one of those options we will see later with other function



```
def iniciarCuento(self):
    canal = pygame.mixer.Channel(2)
    sonido = pygame.mixer.Sound(os.path.join("sonidos", "inicio.wav"))
    opcion1 = pygame.mixer.Sound(os.path.join("sonidos", "opcion1.wav"))
    canal.queue(sonido)
    canal.queue(opcion1)
    self.proceso_de_eleccion(canal)
```

Another important function is "eleccionAcontecimiento". This function is very important because it is the one that allows us to decide between option 1 or option 2, this function give us the two options that Lucas has for continue his journey.



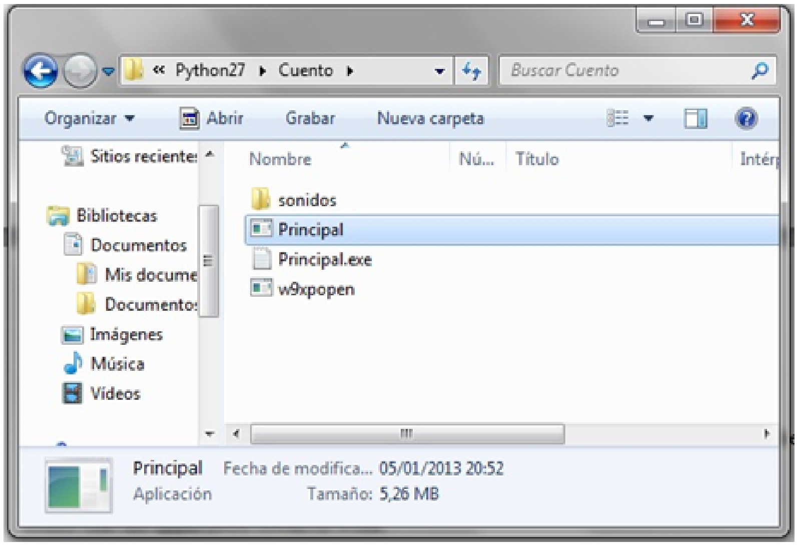
```
def eleccionAcontecimiento(self,titulo1,titulo2,opcion_acontesimiento1,opcion_acontesimiento2,canal):
    verificador = False
    while not verificador:
        for evento in pygame.event.get():
            if evento.type == pygame.KEYDOWN:
                if evento.key == pygame.K_1 :
                    self.detenerSonido(canal)
                    acontecimiento1 = pygame.mixer.Sound(titulo1)
                    canal.queue(acontecimiento1)
                    self.verificar_existencia_opciones(opcion_acontesimiento1, canal)
                    verificador = True
                    r=1

                if evento.key == pygame.K_2 :
                    self.detenerSonido(canal)
                    acontecimiento2 = pygame.mixer.Sound(titulo2)
                    canal.queue(acontecimiento2)
                    self.verificar_existencia_opciones(opcion_acontesimiento2, canal)
                    verificador = True
                    r = 2

            if evento.type == pygame.QUIT:
                exit()


    return r
```

2.6. Manual Of Operation

- - To Use the Application: To use our application is very simple because we have created an executable file to facility of the user; the only thing that we have to do is to have this executable file inside the same folder where we have the folder called "sonidos" that contains all the sound that the application needs to work

- - Starting to play: So to start playing we just have to make double click on the executable file and a window will be opened. Then we just have to listen the story and choose the options that we think that they are the best.

- - Deciding: After we have listen the two option that Lucas has to continue his journey, to choose one of those options the only thing that we have to do is press the number one key or number two key depending on the option that we want to choose. It is not necessary to wait until the narration ends to choose one of the two options, we can make our decision while we are listening the narration



2.7. Observations

Comparing Java vs. Python. Java is one of the programming languages more used by us and we have already develop some applications using this language, so we think it would be good to make a comparison between Java and Python that is the new language that we have learned.
One of the first differences that we notice between Java and Python is that in Python we don't have to put the body of a function between brackets, we just define the function and at the end we put two point (:).
One thing that is really important in Python is the indentation, as I said before here we don't use brackets so how do we know what is part of a function? Because of this question the indentation is so important.
In Python we have to make tabulations to say that something is part of a function, so if we have mistakes with the indentation our function could do something that we don't want to

do. Another think that I like about Python is that you don't have to put the data type of a variable when you define it; you just have to put a name and give it a value. Something similar happens with the definition of function because you don't have to put the data type that the function will return, you just have to put the word "def" before the name of the function, then the parameters between parentheses and that's it.

2.8. Conclusions

Here there are some conclusions that we could make after programming with Python:

- Definitely programming with Python is faster than programming with Java, of course if you learn to use it well. Python is faster, simply because we write less code than we write in Java
- It could be a little confuse when using Python at first for some people, because we are used to program in other languages a little different

2.9. Experiences

To develop this application we had to follow a long process, because this goes from create the story to link all the ways that Lucas could take to reach his goal.
To create the story we have to read some other stories, I think this was the easy part of the project, a little harder was link all the ways to the different ends that the story has. Another thing that I enjoy was when we had to record the narration of the story because it was funny to listen to our partner narrating the Lucas' journey and putting the correct intonation and that kind of thing.
But we had problems when we had to upload our project to GitHub because our folder that contains all the sounds of the story had a weight of 169MB and it was too heavy to upload it. I think that it was pretty easy when we had to start to program because we had already created our story so the only thing that we had to do was translate everything to Python code. Working with sounds was a little complicated at first, because we didn't know how to control the sounds in such a way that two sounds don't sound at the same time.

3. Title of the project: Lucas' Adventure

3.1. Authors

* Mira Rodríguez Raúl Alberto
* Romero Triviño Jose Andrés
* Maya Herrera Ricardo David

3.2. Introduction

Our projects is based on the implementation of a specific algorithm called ”Hill climbing algorithm” to be able to solve the famous game Master Mind. Although many implementations exist not only of this algorithm if not of many other facts in “Java“, Ç”, “C ++” etc . . . Our group had the task of implementing it in a very peculiar and new language because it is the first time that we experience in a language purely functionally. We are speaking about “Haskell“, which although at first it was a little difficult to be able to understand the implementation and syntax of this one, we manage to fulfill our principal targets that were according to our needs:

- To be able to achieve that the machine with a series of steps or processes could guess the code that we have in mind.

- To be able to learn a way different from the one that we had till not long ago of programming, to manage to solve problems or projects of a simplest and effective way contrasting with other languages that we know and selecting the one that more adapts itself to our needs.

3.3. Brief history of Master Mind

Master Mind was invented in 1970-71 by Mordecai Meirowitz, an Israeli Postmaster / Telecommunications expert. His idea was at first turned down by many of the leading toy companies, but he persisted, and took it to the International Toy Fair at Nuremberg in February 1971, where he showed it to a small English company, Invicta Plastics Ltd. The small Leicester based company bought up the entire intellectual property rights to the game, and under the guidance of it's founder Mr. Edward Jones-Fenleigh, refined it, and released it in 1971-72. It was an immediate hit, and went on to win the first ever Game of the Year Award in 1973. It also received a Design Centre Award, and the Queen's Award for Export Achievement. Something about this game caught the imagination of the public, and it became the most successful new game of the 1970's. It has sold over 55 million copies in 80 countries around the world. Mastermind is still being made and sold today.

Link: http://www.tnelson.demon.co.uk/mastermind/history.html

3.4. MasterMind and the Scientific reasoning

The skills of scientific reasoning can improve with the training, but it is difficult to be able to give it like abstract concepts in the classroom. Here, there discusses the possibility of using the game of Mastermind as tool to help the students to develop his logical skills, the design of effective experiments, and to discuss the scientific reasoning in the classroom or the laboratory.

The video game decoder, known as Mastermind, has been adapted to be applied in fields like the mathematics, the computer science and the psychology. The Mastermind proposed like tool for the education of the logic in the mathematics courses, although his accented skill to solve problems also makes it excellent for the life sciences. We propose that to be able to be used the game to teach concrete lessons and to generate discussions on the scientific reasoning, including topics like the experimental sound design, testing hypothesis, to take care of the results interpretation, and of the effective use of the control panel.

In certain aspects, the game simulates an experimental research project, since it is possible to play concerning minutes, without any cost. It enables a language advanced, before to the scientific formation, and does not need laboratory facilities.

Briefly, the çodifier"believe a secret code, which the "decoder"tries to discover in few shifts as it is possible. In the examples that happen here, the code is a sequence ordained as four colors, selected from six possible colors: Red, Blue, Green, Yellow, Orange and Pink.
The Decoder obtains a track on the code (that is to say, it realizes an experiment), interprets the information provided by the Codifier (that is to say, the result of the experiment), and uses this information to design the following experiment. Because to win depends on the reduction from 6 to 4, which supposes approximately 1.296 possible solutions to 1 in the minor possible number of experiments, the logical reasoning and the good experimental design are essential. The game, therefore, provides a simple and practical frame for the discussion and the practice of important scientific skills. And also to play is entertained.

Link: http://bitnavegante.blogspot.com/2011/01/el-mastermind-para-practicar.html

3.5. Analysis of the algorithm

The following analysis and steps that we are going to implement are extracted of the Department of Computer Science of the University of Bristol which says the following thing to us:

- We submit to the Code maker a random guess constructed with 4 genes that we call the "Current Favourite Guess" (CFG).

- From the CFG, we induce a new potential code.

- If submitted guess scores [0,0] then suppress from the pool of colours all the colours present in the last submitted guess. Then find a new random combination (with the new pool of valid colours) consistent with all previous guesses' scores and set this new combination as new CFG and submit it to the code setter.

- If submitted guess score is as good as or better than CFG score, then set this guess to be our new CFG and also set the new score as best score.
- If submitted guess scores [4,0], stop otherwise go to step 2.

3.6. Induction of new potential guesses

The idea behind our stochastic combination generator is to produce new combinations directly from the guesses (and their respective scores) which have been submitted throughout a game of MM. This method, unlike Bento's, does not use any complex formulas and consequently produces new potential combinations with less computation (less processing power). This new method has other advantages described at the end of this section. Let's take an example and imagine a game where:

• The code to discover is "2413".

• The first guess we submit is "1233" and we get [1,2] as score returned by the code maker. "1233" becomes our CFG.

The generation of a new potential code with the method consists of three steps:

Step 1: The number of pegs to keep from our first guess/CFG depends on the first figure between square brackets which is equal to one in this example. So in this case we apply step 1 only one time and if the 2nd digit is randomly selected then the peg to keep is "2" in the second position. Our partial potential combination becomes "x2xx".

Step 2: Pegs to shift depends on the second figure between square brackets, which is 2. So we repeat step 2 two times. In this example, let's suppose the 1st and 4th digits are randomly selected, so we shift "1" and "3" to new and unoccupied locations in the partial potential combination and obtain "321x".

Step 3: The 4th digit "3" from the initial combination needs to be mutated by a new digit with a value different from "1", "2" or "3". This peg can be selected randomly, but this step does not work for all scores. For example:

We imagine a guess code that is "1123" and we subbmit a score tha is [0,3]. The code breaker has to guess the new code with the last score implementing the step 2 so, imagine that the code breaker select the numbers "1", "2" and "3". If in this case we use the step 3, the new potential code is formed for this numbers, and the last number that the code breaker has to choose accordind with this step has to be different that the numbers in the potential code but we have a problem beacuse that number has to be "1" according with our guess code but according with this step that number has to be different of "1". For this reason we have to be more careful in the implementation of this process.

3.7. Implementation of the algorithm in Haskell

- Function ”to generate combinations”who entrusts itself to create all the possible guesswork of 4 digits.

```
generar_combinaciones :: [([Int],Int)]
generar_combinaciones = let
                         numeros =[1..6]
                         combinaciones = [(([w,x,y,z],0)|w<-numeros,x<-numeros,
                                                          y<-numeros,z<-numeros]
                         in combinaciones
```

- Principal function step 2 which is in charge of realizing the step 2 of the algorithm. This function receives the potential code, the positions of the numbers that it is going to change, the numbers of the positions to which it cannot move them, the arrangement of tuples that will return me with the number and his new position which at first sends it to itself as empties and finally an iterator.

```
principal_step2::[Int]->[Int]->[Int]->[(Int,Int)]->Int->[(Int,Int)]
principal_step2 cfg random incorrect_position finish_array 5 = finish_array
principal_step2 cfg random incorrect_position finish_array iterador =

if iterador ‘elem‘ random
 then let new_pos = new_position iterador incorrect_position (1)
          new_incorrect_position = new_pos:incorrect_position
          in principal_step2 (tail cfg) random (new_incorrect_position)
                             ((head cfg,new_pos):finish_array)(iterador+1)
 else principal_step2 (tail cfg) random (incorrect_position) finish_array(iterador+1)
```

- Function new position that is used by the function previously renowned. It receives the position in which I am and it decides which position I move, according to the list of positions to which it could not mobilize me.

```
new_position :: Int->[Int]->Int->Int
new_position actually_pos prohibited_pos 5 = 0
new_position actually_pos prohibited_pos contador =

if not(contador ‘elem‘ prohibited_pos) && contador /= actually_pos
    then contador
    else new_position actually_pos prohibited_pos (contador+1)
```

- Principal function step 1 which is in charge of realizing the step 1 of the algorithm. This function receives the potential code, the positions to be supported according to the algorithm, an iterator and finally the tuples arrangement with the number and the position to be supported.

```
principal_step1::[Int]->[Int]->Int->[(Int,Int)]->[(Int,Int)]
principal_step1 [] random_pos iterator  array_result = array_result
principal_step1 cfg random_pos iterator  array_result =
if iterator ‘elem‘ random_pos
   then principal_step1 (tail cfg) (random_pos) (iterator+1)
                                     (((head cfg),iterator):array_result)
   else principal_step1 (tail cfg) (random_pos) (iterator+1)  (array_result)
```

- Function verify for step 3, which verifies to me that the number or the numbers given randomly by the machine are not in my new possible potential code. It is necessary to stress that this function does not happen for the score [0,3] due to the already said earlier.

```
verify_for_step3 ::[Int]->[(Int,Int)]->Bool
verify_for_step3 array [] = True
verify_for_step3 array array2 = if fst(head array2) ‘elem‘ array
                                 then False
                                 else verify_for_step3 (array) (tail array2)
```

- Principal function step 3, which already with the numbers previously verified, it places them in an arrangement of tuples, just person with the positions that need and 1 or 2 have not been used in the step.

```
principal_step3::[Int]->[Int]->Int->[(Int,Int)]->[(Int,Int)]
principal_step3 [](incorrect_position) (iterator) (array_result) = array_result
principal_step3 (select_numbers) (incorrect_position) (iterator) (array_result) =

if iterator ‘elem‘ incorrect_position
   then principal_step3 (select_numbers) (incorrect_position) (iterator+1)
                        (array_result)
   else principal_step3 (tail select_numbers) (incorrect_position) (iterator+1)
                        ((head select_numbers,iterator):array_result)
```

- Function result for new cfg, which is in charge of generating the new arrangement or potential code. It receives the arrangements of tuples given by the previous functions of the step 1, step2, step 3 according to the dedicated score.

```
result_fornew_cfg :: [(Int,Int)]->[(Int,Int)]->[Int]->Int->[Int]
result_fornew_cfg [] array2 result_Array iterator = result_Array
result_fornew_cfg array1 array2 result_Array iterator =

 if snd(head array1) == iterator
    then result_fornew_cfg (array2) (array2) (result_Array++[fst(head array1)])
                           (iterator+1)
    else result_fornew_cfg (tail array1) (array2) (result_Array) (iterator)
```

3.8. Remarks

Along the whole achievement of our project we have met certain difficulties that we want to mention in this document which we'll mention them next:

- Indentation:

  On having begun with our project, we had many problems with regard to the indentation in this language, it can be because it is our first scripting language in contrast to python which in spite of also it being, we could realize it in eclipse by means of a pluggin. But due to many factors we could not do it the same way with haskell, so we resort to notepad ++ since it was that with more we are familiar. But on having begun programming, we find many errors of compilation which mainly it belonged to indentation, an example was that, on having given enter to continue in the following line and to keep on programming, in the compilation it was showing error in this line, the only solution was to erase all the spaces before the enter and to be able to come again to the same line only with spaces. This type of errors was taking a lot of time to us to find it if we had already written too much code.

- Random numbers:

  We had also a little of difficulties with the random numbers that the algorithm needs in his implementation. On having wanted that this function works in dynamic form, we could not realize it this way because to be able to select random numbers, we make use of a block "do", and any value that we return inside this block, it is of type IO (), that is to say if we want that he returns us a integer, this block there return us an IO Int that is very different from an Int in which procedures cannot be realized like comparison, adds up, remains etc. like a normal integer. Our solution was to work the whole project inside this block "do"because we notice that if this number is not returned, but it is used inside this block works perfectly like a normal integer.

- Different Random numbers:

  On having wanted that in our code there is done a ratification of which these random numbers are all different ones of the 1 from 4, in certain cases we realized that the machine was remaining in an infinite bond on having delivered numbers in which at least one was equal to other, specially this alone happened when the score given by the code maker was [1,3], we do not know because in himself this alone was happening for this certain score, since we check repeatedly the code and everything was perfectly, so the only solution was to create another separate function to select the random numbers for this specific score like that the machine return equal numbers.

- Step 2 for the result [0,4]:

  The problem in this step was that the function that we realize so that all the numbers change position was that regrettably a number was always staying in the same original position. So we realize other one that there was receiving an arrangement [m, n, o, p], in which all his numbers are different and go from the 1 to 4 indicating the positions and it returns an arrangement with different order of the form [or, m, p, n]. Let's suppose that the entry arrangement is [4,1,3,2] and returns [3,4,2,1] specifying that the number that earlier position was going in 4 now anger in 3, that of the 1 changes to 4, that of 3 goes to 2 and 2 goes to the 1.

3.9. Conclusions

We have managed to develop a program programmed in haskell that represents to the famous game master Mind using the algorithm of "Hill climbingüsing a series of steps in functions, so that the computer manages to 'foresee' the code proposed by the code maker, all this handling new elements like lists, tuples, list of tuples, new data types, and a new way of implementing the functions, since now we face a language purely functionally, differently from the imperative languages that we had studying such like C or java.

3.10. Experiences of the development

At the beginning trying to learn haskell, like a new language, and a new form of definition, and implementation of the functions, since it is languages functionally, was like learning a language completely different from those that previously we study, we base on a series of tutors, principally on one called: 'Aprende haskell por el bien de todos', which is the good one, he guides for the beginners on this language: aprendehaskell.es/, already in the development of our project, commonly some disadvantages appeared, with the hardware that we handle, initially, we use notepad ++, and the compiler of CHC, who gave errors especially with some bookstores between them those of random, use a program called WinHugs to compile the functions 'hs' which solved these mentioned problems, another disadvantage was in the handling of the tabulation in notepad ++, who reads the lines in rule.

In the development of our project most of our functions used the methodology of trip of the lists in recursive form, another experience to commonly used like the block for, or for each, in short, in spite of being haskell, a language different from the studied ones, the beginning of programming like the recursion, the declaration, and the bonds have not got lost, for which, we could have learned in a rapid way to develop in this language and power to realize our project mastermind in a correct way.