Natural Science Learning Unit Report

Lucia Fores[a]
[a]fores.1836451@studenti.uniroma1.it

May 21, 2023

1 Introduction

The aim of the record is to describe the process thanks to which I created and built a Learning Unit in ***MIT App Inventor*** whose objective was to teach about **Biodiversity** as natural indicator and to give student a framework thanks to which they could write their own **researches about natural species**.

2 Prerequisites and Reference Target

2.1 Reference Target

My idea is to propose this Learning Unit to a **second year high school natural science's class** (for the biology module), so to 15/16 years old students.
This decision was made mainly thinking about the Italian School Ministerial Program for the high schools (specifically for the **Scientifico** or **Scienze Applicate** addresses): in the second year of this school cycle, for the **Natural Science** classes, the student are asked, for the biology module, to learn about the classification of the living being in five different classes (**kingdom monera**, **kingdom protista**, **kingdom plantae**, **kingdom fungi** and **kingdom animalia**) together with the deepening of the characteristics of each kingdom.
So the idea was to give the students a way of collecting all their knowledge about three of the five kingdom (only the plantae, fungi and animalia) and to "apply" the data collected to calculate the natural indicator of biodiversity (so to give them an example of how much is important to exploit the notion learned in class to better understand how real life works).
It was also decided to give the student a basic knowledge about how much the presence of **native, allochthonous or endangered** species can **influence the biodiversity** of a place.

2.2 Prerequisites

The prerequisites required are the following:

- **Biology**
  - Cell theory
  - Differences between prokaryotes and eukaryotes
  - Differences between heterotrophs and autotrophs
  - The origin of complex life: the evolution of the eukaryotes and the evolution of the plants
- **Math**
  - Operation on natural numbers and specifically about divisions
  - Concept of ratio between two different quantities
- **Computer Science**
  - Notion of variables
  - Notion about more complex data structures especially lists and dictionaries
  - Notion of procedures and functions
  - Notion of conditions and loops
  - Notion of databases

3 Learning Objectives and Project Guidelines

3.1 Learning Objectives

At the end of the Learning Unit the students should have a good knowledge about the MIT App Inventor framework, the block-procedural and the event-based programming style, the way to design a mobile application, how different parts of the application communicate together (specifically the front-end, back-end and some interaction with the user) and how to design a No-SQL database (better if they are able to pass from the design to the implementation by themselves).

On the Natural Science side they must at least know the division of species into the three most generic kingdoms (so at least divide them into Animals, Vegetables and Mushrooms) and also know how to provide a basic computation of the Biodiversity (better if they can also be able to explain the impact of the presence of the different species - meaning native, allochthonous or endangered - on the biodiversity of a place).

3.2 Project Guidelines

The following guidelines will be given to the student at the project assignment:

"You have to build, using the MIT App Inventor system saw during the classes, a mobile application representing a Natural Journal which you will use to collect observation about the natural species surrounding you. It is important to define a system to collect the data about the species and to provide a way (even if it will be a simple one) to compute the biodiversity index representing the healthiness of a place. Notice that you can use the provided formulas for the computation of the biodiversity. For the evaluation of the comprehension of the MIT App Inventor system it is required to provide at least a working mobile application (with both front-end and back-end) that also permit the user to use some of the sensors saw in class. As for the Storage part I recommend you to use the TinyDB extension saw in class and to think about how structure your data information by looking at the example of key-value database provided or the ones we will see during final classes"

The materials given to the students are the following:

- **Biodiversity Formula**

  $$\frac{SpeciesRichness}{SpeciesEvenness}$$

  This is the minimal formula [1] that can be used to compute the biodiversity index, it will be then a task for the students to understand how connect the notion of different types of species with the notion of Biodiversity to update the formula (which can still be used but it will not be enough for the maximum grade)

- **Database Entry Example**
  <tag_name> : <value>
  Notice that the value can be whatever data structure you like, also the more complex one (like lists and dictionaries) in order to represent more complex information (e.g. you want to store all the characteristics of a specie, you could provide something like
  <scientific_name> : <data_structure_containing_all_the_specie_characteristics>

- **Front-End Extension with Example**
  An extension [2], and an example of its usage, to make the front-end of the application more pleasing.
  This extension give the user the possibility to specify paddings and margins for each component of the screens, since it is something optional to be used, the students will also have an example (made by the creator of the extension) that explain all the main features that can be used but it will be their duty to study it and understand it

4 Class Structure

The Learning Unit will be taught during the final months of the third year: in this way the students will have a basic knowledge about programming (scheduled for the early year).
The Learning Unit will be developed in 8 lessons since this amount of time would be enough to present the needed features of MIT App Inventor for the project (and the Natural Science concepts to be implemented in the project): specifically the first three lessons will be used to present the main features of the system (precisely the event-based structure of the application, the way in which the user can interact with the application through the sensors and a general explanation of the storage systems that MIT App Inventor offers); then during fourth and fifth lesson the biology knowledge required for the project will be presented, meaning the distinction in kingdoms and the biodiversity index. At the end of the fifth lesson, when the student will have already a knowledge about the App Inventor system and about the biology notions

needed for the project, the project guidelines will be given to the students. During the last three lesson the student will continue the project by also having the possibility to follow some examples of how to handle similar situation in App Inventor presented in class (mainly focusing on how to save and which structure give to the data when saved in the database); during the last lessons it will also be presented the concept native, allochthonous and enraged species so to insert this concepts in the application.
At the end of the eight classes a lesson will be used to give them the possibility to present the project and evaluate their work.

5 Evaluation Grid

Descriptors	Mark
The student owns a wide, complete and deep knowledge on the topic. Specifically they can demonstrate to have learnt: how correctly divide the species into the kingdoms, how systematically and scientifically classify new species by providing the correct amount of notions to describe them, the concept of biodiversity, the correlation that exists between the presence of the species in an ecosystem and the corresponding biodiversity, the basis needed to develop a full mobile application and to correctly and technically explain the difference among back-end, front-end and all the parts that compose them, how to design a graphically appealing application, the difference between a relational and a non-relational database and how to develop a non-relational database. The students also have been able to correctly integrate the concept of native, allochthonous and endangered species in the computation of the biodiversity and were able to provide their personal contribute to the project amplifying it with their personal researches and knowledge	Excellent
The student owns a complete and deep knowledge on the topic. Specifically they can demonstrate to have learnt: the correct classification of the species into the kingdoms, how to scientifically describe a new specie, the concept of biodiversity and the correlation that exists between the presence of the species in an ecosystem and the corresponding biodiversity, the basis needed to develop a full mobile application, how to design a graphically appealing application, the difference between a relational and a non-relational database and how to develop a non-relational database. The students also have been able to correctly integrate the concept of native, allochthonous and endangered species however the project lacks on the personal contribute given by the students about personal researches and personal notions integrated into it	Optimal
The students own complete knowledge on the topic. Specifically they can demonstrate to have learnt: the division into kingdoms and how describe a specie, the concept of biodiversity, the basis needed to develop a full mobile application, the difference between a relational and a non-relational database and how to develop a non-relational database. The students also have been able to correctly integrate the concept of native, allochthonous and endangered species in the computation of the biodiversity. However the project lacks a graphically pleasing front-end.	Good
The students own a barely sufficient (both in completeness and correctness way) knowledge about the topic. The students can demonstrate to have learnt: the division into kingdoms and how to describe a specie, the concept of biodiversity, the basis needed to develop a full mobile application, to be able to develop a non-relational database. The students did not integrate the concept of native, allochthonous and endangered species in the computation of the biodiversity.	Sufficient
The student owns incomplete knowledge on the topic. In their project they weren't able to demonstrate their comprehension of how to structure the information about the species (both in a naturalistic way, meaning they weren't able to identify the crucial information needed to describe the specie, and in a computer science way, meaning they weren't able to find a way to represent the data with a data structure or find a proper way to design the database) leading it to a wrong computation of the biodiversity index.	Mediocre
The student owns no from to wrong knowledge. They weren't able to produce a working project demonstrating a big gap in their comprehension of the presented topics meaning they produced a project with big errors both in the execution and design; example errors could be the use of a wrong formula for the biodiversity index, the wrong classification of the species in the kingdoms and the wrong design of the application both in the front-end and in the back-end.	Insufficient

6 Development

6.1 Excellent Development

The excellent development of the project consists in an application of **7 screens**.
Before explaining in detail the screens is useful to give the schema of the database which is represented in Listing 1

```
"tmp_image" : <image_of_a_new_observation>,
"tmp_location" : <location_of_a_new_observation>,
<scientific_name> : {
      "image" : <observation_image>,
      "commonName" : <observation_common_name>,
      "scientificName" : <observation_scientific_name>,
      "family" : <observation_kingdom>,
      "description" : <observation_description>,
      "locations" : [<sighting_location_1>, ..., <sighting_location_n>]
}
"types" : {
      <observation_scientific_name> : {
            <location_1> : <type>
            <location_2> : <type>
            ...
            <location_n> : <type>
      }
}
"evenness" : {
      <observation_scientific_name> : {
            <location_1> : <evenness>
            <location_2> : <evenness>
            ...
            <location_n> : <evenness>
      }
}
"quadrats" : [<location_encountered_1>, ..., <location_encountered_n]
```

Listing 1: Application Database Schema

Specifically the screens have the following functionality:

1. **Home Screen** (Screen 1 in the code, depicted in Figure 1)
   This screen is used to welcome the user in the application

Figure 1: Home Screen of the application

2. **Description of the biodiversity index** (Screen 6 in the code, depicted in Figure 2)
   This screen is used to explain to the user what is the biodiversity and how is computed

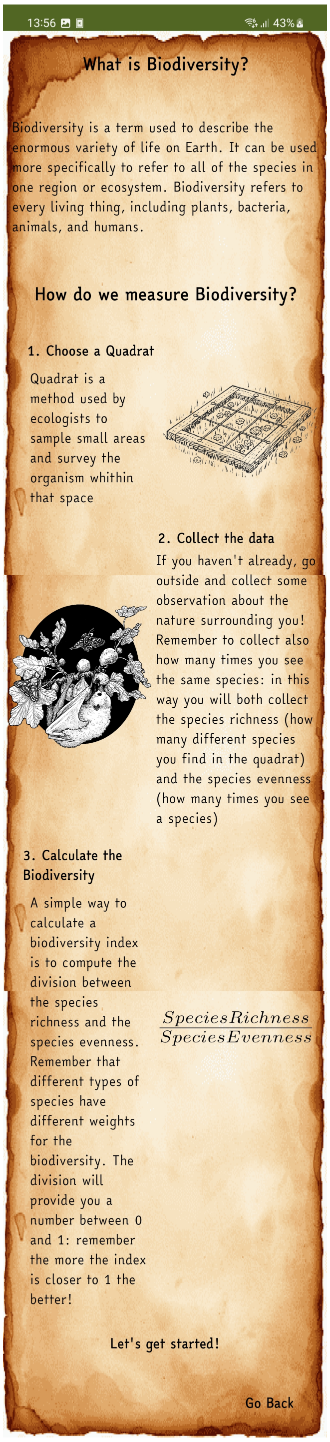

Figure 2: Description of the biodiversity index

3. **User Actions** (Screen 2 in the code, depicted in Figure 3)
   This screen is used by the user to choose which action they want to enter: they can decide to add a new observation, go to the saved observation or go to compute the biodiversity of one of the saved places

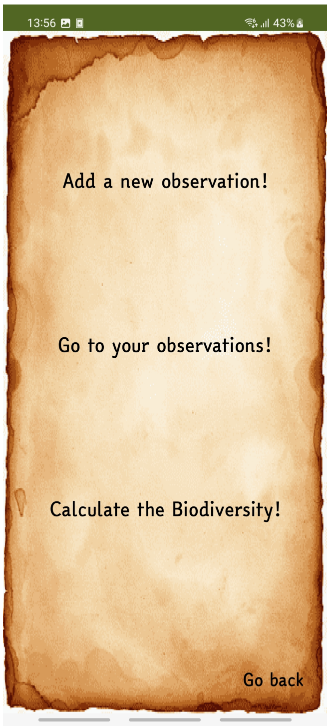

Figure 3: Actions to be entered by the user

4. **Addition of a new observation** (Screen 3 in the code, depicted in Figure 4)
This screen is used by the user to add a picture (by snapping it or by choosing it from the camera roll) of a new observation. Notice that by giving the wanted image to be saved (stored in the database under the tag `tmp_img`), the application will register the user position and will save it as the sighting location of the specie (stored in the database under the tag `tmp_location`): if the position is not recorded correctly (the sensor wasn't accessible or the server didn't respond quickly with the position) this will not be saved and could be insert later by the user

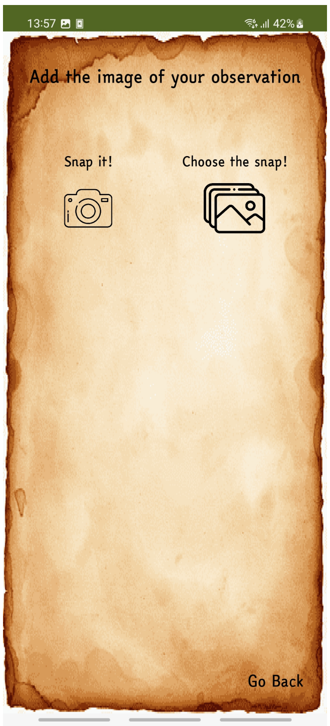

Figure 4: Add a picture of a species

5. **Fill of the observation record** (Screen 4 in the code, depicted in Figure 5)
This screen is used by the user to fill the record for the new observation; it is specifically asked to provide the specie **common name**, **scientific name**, **family** (one among *animal*, *vegetable*, *mushroom*), **description** and the **type** (one among *native*, *allochthonous* or *endangered*), the record will also display the location saved previously.
At this point the data will be stored in the database: specifically (as reported in Listing 1) it will be created a tag with the scientific name of the specie and all the data will be reported under the specific tags

Figure 5: Specie's Report during the filling action

6. **Visualization of the observations** (Screen 5 in the code, depicted in Figure 6, Figure 7, Figure 8)
This screen is used to let the user decide for which observation made they want to visualize the record

Figure 6: User saved observations

Once chosen the observation the record is presented to the user, at this point the user can decide to look at the statistics (evenness and type) of the specie for a given place (Figure 7, notice that by choosing a place the type of the specie will appear and it will be given to the user the possibility to update the evenness of the specie in that place) or, by clicking on the button `Add a new sighting place!`, they can add a new place in which the observation has been made and they can choose the type of the specie for that place (Figure 8).
If the evenness is updated than the application takes care to update the tag for the specie in the tag `evenness`, if it's registered another location instead the application takes care of check if the location is a totally new one (and if it is the location will be added to the list saved under the tag `quadrats`) and to update the data under the tag `locations` of the tag with the scientific name of the specie; also the data under the `evenness` tag will be updated by putting the evenness for the specie in that location to 1 (which is the default value that can be updated later) and the data under the `types` tag will be updated by adding a new key-value pair with the new location of the specie and the type registered for the specie

Figure 7: Observation Record

Figure 8: Add of a new sighting place

7. **Computation of the biodiversity index** (Screen 7 in the code, depicted in Figure 9, Figure 10) This screen is used to give the user the possibility to choose the ***quadrat*** (the location) for which they want to compute the biodiversity index.

Figure 9: List of all the saved quadrats

Once the location has been selected the computed biodiversity will be presented to the user with also a textual judge about the biodiversity of the place and a button connected to an *ActivityStarter* that let the user go to a link in which are reported notions that can be used by the user either to maintain a certain level of biodiversity or to restore it.
The *ActivityStarter* will start a `VIEW` action that will lead the user to the linked web page

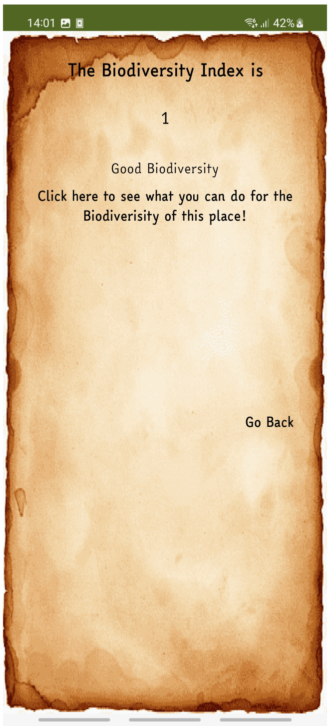

Figure 10: Add a picture of a species

6.2 Sufficient Development

The sufficient version of the project has been developed starting from the guidelines expressed in the evaluation grid: differently from the excellent version in this project the extension for the GUI was not used and the species has been described with a more simplex record (there was not the possibility to insert a description and the division in types); consequently the database structure is simpler and also the formula used for the computation of the biodiversity is the one provided without the integration of the concept of native, allochthonous and endangered species.

References

[1] Maryland Science Center. Calculating biodiversity. https://www.mdsci.org/wp-content/uploads/2020/06/CalculatingBiodiversity.pdf, 2020.

[2] Tomislav Tomsic. Gui extension. https://github.com/Tomislav-Tomsic/GUI-Extension, 2018.