Trent University

Carbon Calculator

Final Project for COIS 2240

Erin Lonergan

4-3-2019

Description of Software:

Our group 23, consist of the team members Erin lonergan and Maxime Villeneuve. we chose to create a software that would not only be beneficial to this course but to the world. Our topic is a carbon footprint calculator that is used to give users an understanding of just how much their life style and everyday routines contributes to CO_2 emissions and the harmful environment impacts endured by planet Earth. Our Software monitors the average individuals carbon footprint based on there day to day life routine and compares their average to the national carbon footprint Average generated by our database. This software will have multiple questions that it asks the user upon start-up of the software, from which the users input replies to the given questions. The application will calculate the user's results by turning the answers given by user into a numerical value-based evaluation scheme comparing it to the questions numeric value of importance/ impact on the environment. From there, a calculation is made and then is outputted to the user along with the Nation average, so user can compare if their carbon footprint is above or below or equal to nations average. There will also be an evaluation legend at the end of test that shows user an example of a good or bad score as far as how much or little their lifestyle contributes to CO_2 emissions and negative environmental impacts.
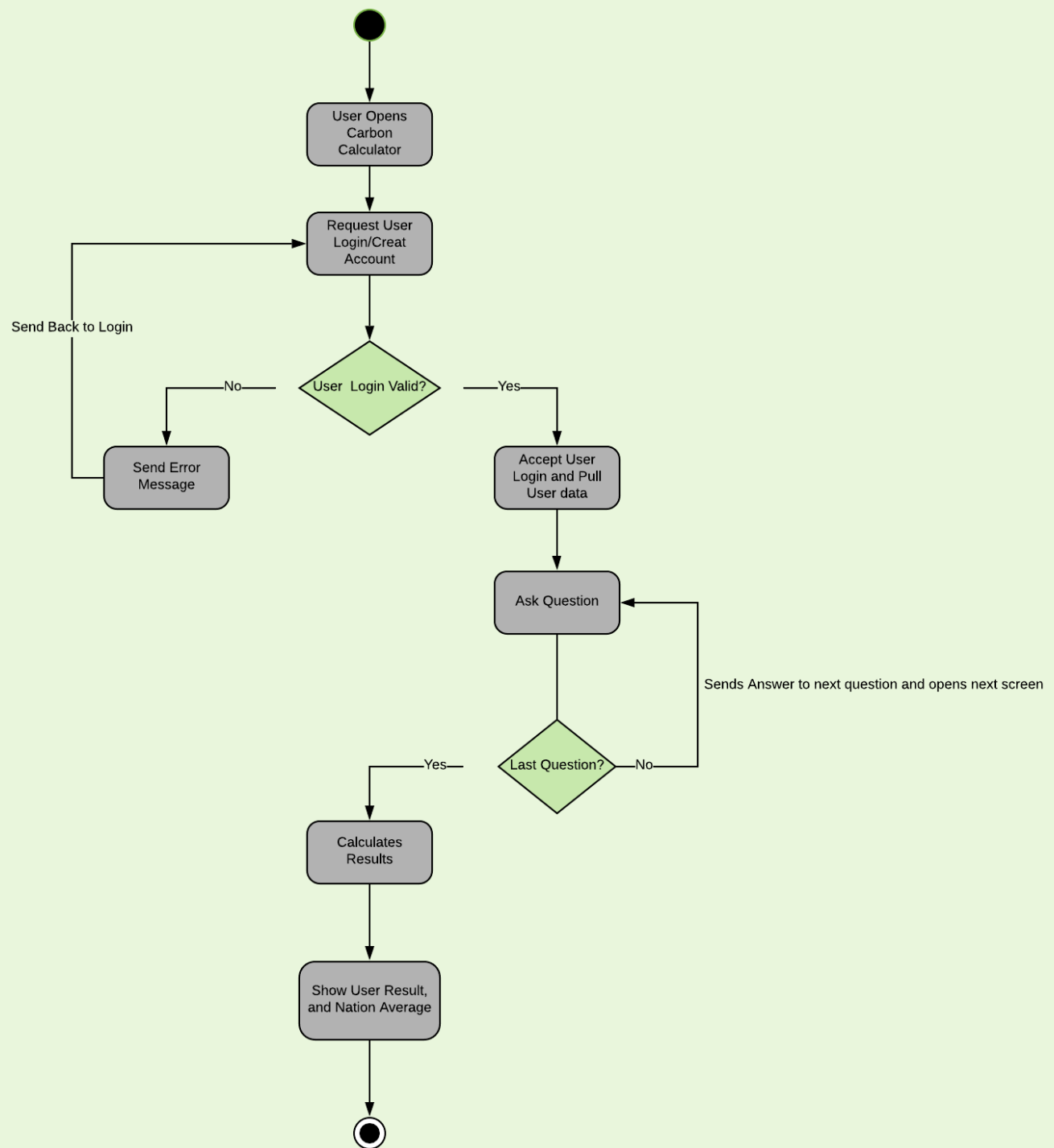
Target Market & Industry:

The target market for this innovative idea is millennial generations and younger. Having an application that can monitor, your carbon footprint, and environmental impact is an application that can be very appealing to the new and environmentally aware generations. This Software targets the Environmental industry that are concerned with the economic and human impact that is causing detrimental effects on our planet. To environmental industries this software can be used as a way to not only gather more data about societies lifestyle impacts that are harmful to the planet but also becomes appealing to consumers to take test such as this one and ultimately getting society more interested in protecting our planet.
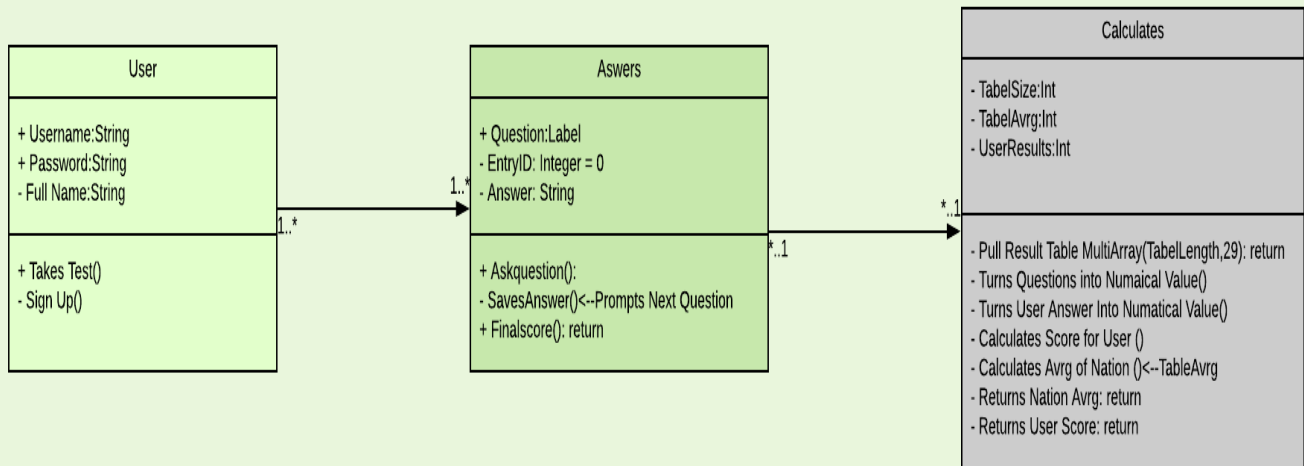
Functionality Details:

This program will implement a GUI to let people input answer the questions. It will feature a series of pages that will each ask their own question (29 in total). All the answers will be stored in a database and at the end calculate the carbon footprint of the user and compare it to the average of the database.

```
                    ●
                    │
                    ▼
            ┌───────────────┐
            │  User Opens   │
            │    Carbon     │
            │  Calculator   │
            └───────────────┘
                    │
                    ▼
            ┌───────────────┐
            │  Request User │
            │  Login/Creat  │ ◄──────────────┐
            │    Account    │                │
            └───────────────┘                │
                    │                        │
                    ▼                        │
              ◇ User  Login Valid? ◇         │
          No─┘                  └─Yes        │
          │                        │         │
          ▼                        ▼         │
  ┌───────────────┐        ┌───────────────┐ │
  │  Send Error   │        │  Accept User  │ │
  │    Message    │        │  Login and Pull│ │
  └───────────────┘        │   User data   │ │
   Send Back to Login      └───────────────┘ │
                                   │          │
                                   ▼          │
                            ┌───────────────┐ │
                            │  Ask Question │◄┘
                            └───────────────┘
```

User

+ Username:String
+ Password:String
- Full Name:String

+ Takes Test()
- Sign Up()

Aswers

+ Question:Label
- EntryID: Integer = 0
- Answer: String

+ Askquestion():
- SavesAnswer()<--Prompts Next Question
+ Finalscore(): return

Calculates

- TabelSize:Int
- TabelAvrg:Int
- UserResults:Int

- Pull Result Table MultiArray(TabelLength,29): return
- Turns Questions into Numaical Value()
- Turns User Answer Into Numatical Value()
- Calculates Score for User ()
- Calculates Avrg of Nation ()<--TableAvrg
- Returns Nation Avrg: return
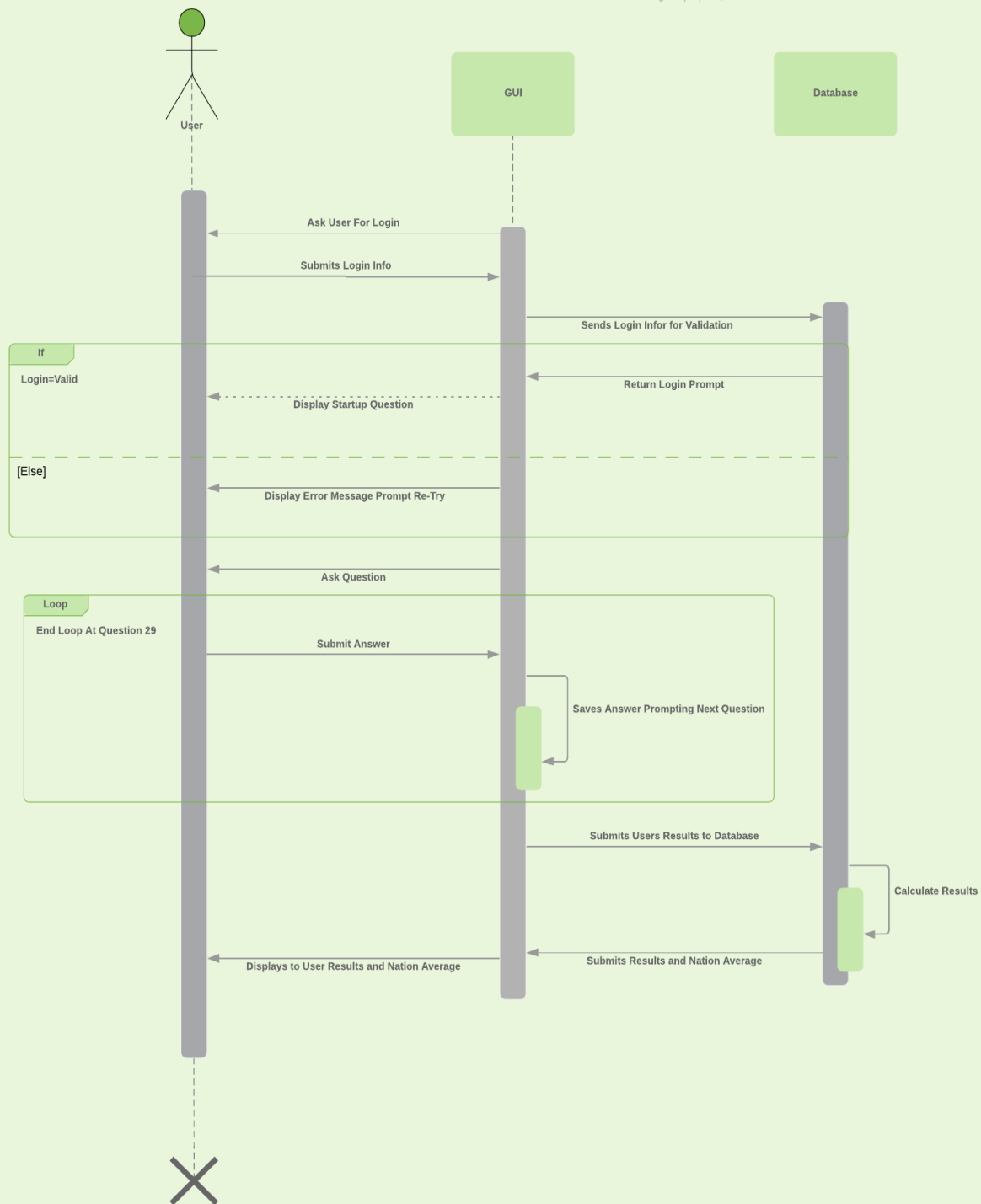- Returns User Score: return

1..*   1..*   *..1   *..1

The user inputs their credentials (username, password, full name) and when they are correct, it takes them to take the test. The program asks 29 questions and in each question store the answer (all well as username on login) in itself in a string array and then passed the array to the next question. There is 2 calculations that happen. One is for the individual user (userResult) and the other is the average of the database (tableAvg). The user's answer is callcaulated by using the last entry in the Answer Table, while the table average creates a 2d array to mimic the database and then procedes to average out. Both of these are converted from string arrays to interger arrays with each answer given a specific weight. Both results are then returned.
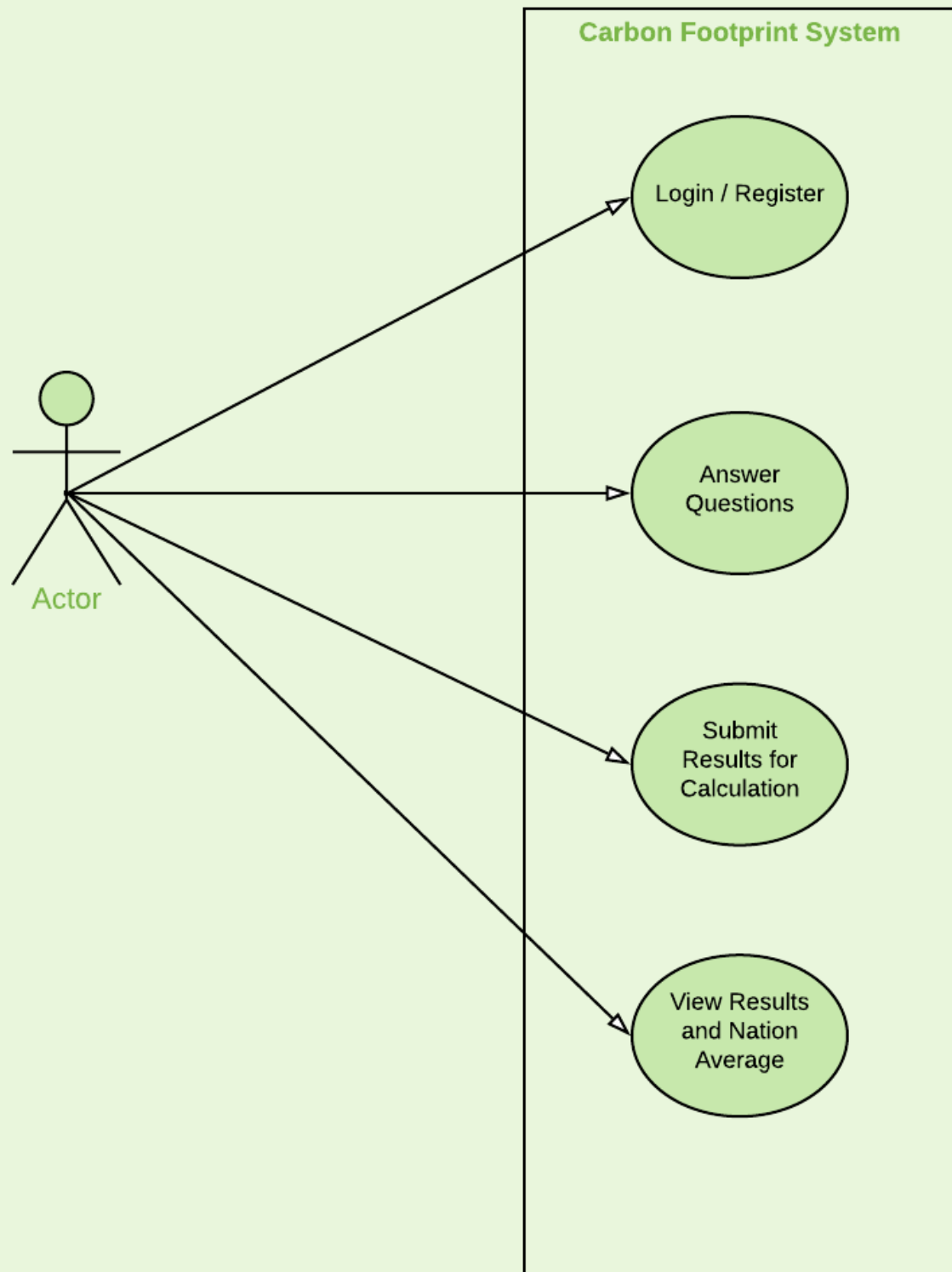
Sequence Diagram

erinlonergan | April 5, 2019

User — GUI — Database

- Ask User For Login
- Submits Login Info
- Sends Login Infor for Validation

If — Login=Valid
- Return Login Prompt
- Display Startup Question

[Else]
- Display Error Message Prompt Re-Try

- Ask Question

Loop — End Loop At Question 29
- Submit Answer
- Saves Answer Prompting Next Question

- Submits Users Results to Database
- Calculate Results
- Submits Results and Nation Average
- Displays to User Results and Nation Average

This Sequence Diagram analyses the prosses in which the software takes to communicate from user, to interface, to database. The main stages of this Diagram are to prompt user for login, from here a condition need to be met to validate user login, if login verified, ask user questions, if not, send an error message and take user back to login screen. After asking user question, the user submits a response which prompts a loop to be made that stores each answer to the given questions and store the answers given by user each time which prompts the next question to be asked the loop stops when all 29 questions are asked and then the interface submits results to the database which then calculates the user results and then submits them back to the interface which displays results to user along with Nations

Carbon Footprint System

Login / Register

Answer
Questions

Submit
Results for
Calculation

View Results
and Nation
Average

Actor

This User Case Diagram analyses the prosses in which the external user interacts with the software from start to finish. The main stages of this Diagram are that User logs in with there username and password, then the system asks user questions, user answers the question up to question 29, user submits test for calculation, and final the system calculates user results and then displays results to user along with Nations average.

Project Retrospective:

Inregards to the Building of the program, the easiest part of the project was adding the series of questions to the software and the designing of the interface visual appeal. All questions had the same template, therefore we just had to copy paste each question one at a time and replace the variables that needed replacing. The most difficult part of the project was getting JavaFX to switch scenes at the beginning of the coding process. There are many ways to do it and trying to find one that actually worked with this project was near impossible at first. If our group could do anything different, it would be the following: Have a 3rd table that holds user answers, so we don't have to calculate from the entire database every time the user reached the final page. Another thing our group would do different is make the calculation class a lot more efficient at extracting, converting and calculating data. As of now it works via many, many if / if else statements and its very easy to get lost. Major thing we learned was how to work with a database, how to create a graphical user interface program, and to not put images on your interface in Sceen Builder without putting them into the directory of the project.