James Easson

Grade 12

IT PAT Phase 1

Scenario and Scope

The IT topic is to create a program that can aid in saving the planet.

The world is currently experiencing climate change due to greenhouse gas released by industrial processes, power generation and other human impacts. Climate change has caused an increase in more violent weather such as floods and droughts. There has been an effort by many individuals and companies to decrease their carbon footprints.

I propose a program that can calculate the user's carbon footprint and provide guidance on how to reduce this. The program will be able to calculate their footprint for a single user such as a residential home and personal car but also calculate it for organizations such as businesses or schools. This will be done by the user inputting info such as what transport they take such as car model or taxi and how far they travel. It will also use their consumption of energy and ask questions about some big appliances such as stoves and geysers.

The program will then use a dataset to help calculate their carbon footprints such as car and electricity databases and generate an estimated carbon footprint. The program will also offer suggestions on how to decrease their footprint such as better times to bathe or when to travel.

User Requirement

|  |  |  |  |
| --- | --- | --- | --- |
| Type of roles | Roles | Activity | Limitations |
| Single system | User | * Submits data from their lives for the carbon footprint calculation such as kilometer driven, electricity consumed, and car driven. * Receives the output of the calculated carbon footprints in kilograms. * Gives advice to decrease footprint. * Can set goals for the future. * Can re-enter data often to show trends. * Can view historical data over time to see if they are reaching their carbon footprint goal. | * There are no limits besides not being able to change historical data older than a month. |
| Organization | User | * Submits input data from their day or week such as kilometer driven in a company vehicle. * Can change mistakes in their own data input if there was an error. * Displays their personal impact to the organization carbon footprint. * Can set personal goals. * Suggest remedy to climate footprint. * Can view historical data over time to see if they are reaching their carbon footprint goal. | * Can not change data after 3 days after data submitted to prevent change in climate footprint. * Can not view entire companies carbon footprint. |
| Administrator | * Submits data such as electricity consumption and other big emitter such as a generator. * Can view logs of changes to input data to see if people have been changing their data to often. * Displays organization entire carbon footprint. * Can put data into a file for easy use during meetings. * Can edit some mistakes created by normal users. * Receives suggestion to decrease carbon footprint. * Can see trends over time to see if the company is decreasing their total foot print. | * Can not edit data older than 6 months. |

Database Design

Legend

* PK-Primary key
* FK-Foreign key

User table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Keys | Field length | Data type | Comment |
| UserID | PK |  | Autonumber | Primary key. |
| Username |  | 30 | Text | Stores the generated username of the user. |
| First Name |  | 30 | Text | Stores First name. |
| Last name |  | 30 | Text | Stores Last name. |
| Password |  |  | Text | Stores the user’s password for logging in. |
| Carbon Footprint |  |  | Number | Stores current carbon footprint. |
| OrganisationID | FK |  | Number | Show which organization they are a part of. |
| Last Updated |  |  | Date/Time | Shows when the user last updated the field. |
| Admin |  |  | Boolean | States if the user is an admin in their organization. |
| Goal |  |  | Number | Stores the user desired carbon footprint goal. |
| OrganisationMembership |  |  | Boolean | Stores if the user is part of an organization or using the program in their personal capacity. |
| Email |  | 40 | Text | Stores the users email address |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Key | Field length | Data Type | Comment |
| OrganisationID | PK |  | autoNumber | Connects to tblUser. |
| OrganisationName |  | 100 | Short text | Stores the full name of the organization. |
| Total Carbon Footprint |  |  | Number | Stores the Carbon Footprint. |
| Last Time Updated |  |  | Date/Time | Stores the last time the carbon footprint is calculated. |
| OrganiastionGoals |  |  | Number | Stores the companies desired emission goals. |

Organization table

Car table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Key | Field Size | Data Type | Comment |
| CarID | PK |  | autoNumber | Connects to Car table |
| Make |  | 100 | Short Text | Stores the different brand of cars. |
| Model |  | 100 | Short Text | Stores the model of the cars. |
| Emission per KM |  |  | Number | Stores the per kilometer emission of the vehicle |

Car List

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Key | Data Type | Comment |
| CarListID | PK | AutoNumber | Primary Key |
| UserID | FK | Number | Connects to Users table |
| CarID | FK | Number | Connects to Cars Table |
| CurrentMileage |  | Number | Stores the current mileage of the car |

Footprints table

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Key | Data Type | Comment |
| FoorprintID | PK | AutoNumber | Primary Key |
| UserID | FK | Number | Connects to Users table |
| EmissionDate |  | Date/TIme | Stores the date when the record was entered |
| Emission |  | Number | Stores amount of emissions |

Relationships

Referential integrity will be enforced

A screenshot of a computer

Description automatically generated

Data Dictionary

Classes and objects

|  |
| --- |
| Class name: CalculateFootPrint |
| Attributes |
| * FCarID : integer; * FElectricty : integer; * Fkm :integer; * Fusername : String; * FDate :TDate;   + FEmmison : integer; |
| Methods |
| * Constructor Create (FCarID, FElectricty, FEmmison, Fkm : integer; Fusername : string; fDate : Tdate); * Procedure CalculateFootprint(FEmmsion);//calculate the footprint by using the data recieved * Function returnFootprint : integer;/Displays the calculated foot print for the user |

|  |
| --- |
| Class name: SignIn |
| Attributes |
| * Ffirst\_name : string; * Flast\_name : string; * Fusername : String; * Fpassword : string; |
| Methods |
| * Constructor ( Fusername, Fpassword : string); * Constructor (Ffirst\_name, FLast\_name : string); * Function PasswordCheck (attempt:integer): boolean; //checks if entered password is correct * Procedure UsernameGeneration(Ffirst\_name, Flast\_name : string); |

Text files

|  |  |  |
| --- | --- | --- |
| Text file name | Variable name in program | Description |
| Logging.txt | Log | Logs all changes made to the database who made it and when it was made to check for errors or malicious changes to the programs data. This file will become a large file over time. The admin can delete the file or output it in the desired format. |
| PreviousData.txt | Previous | This stores all previous data the program created such as organization previous carbon footprint, personal footprint. It will also store their previous goal to show if they reached their target. |

Array

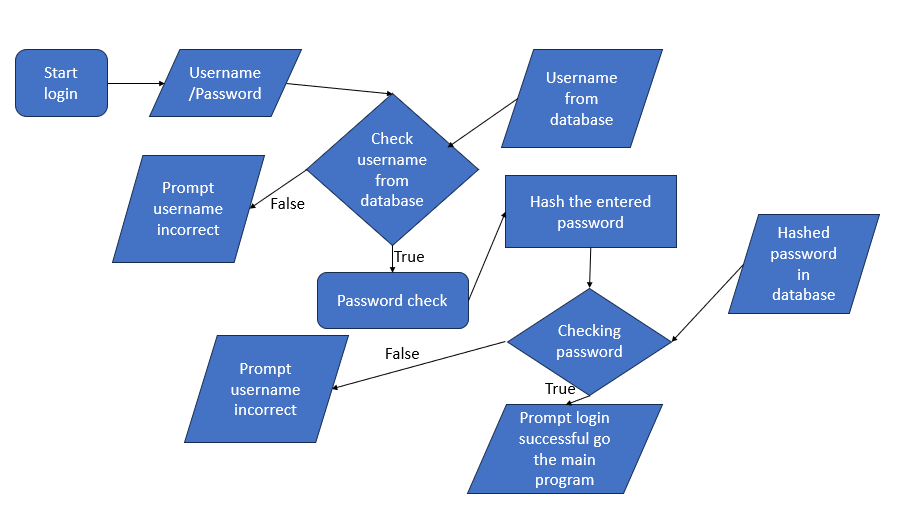
* A one-dimensional array will be used to combine all of the user's carbon footprints into the organization's final value.
* An array will also be used to store the table data until it is written into the text file.

Advanced programming

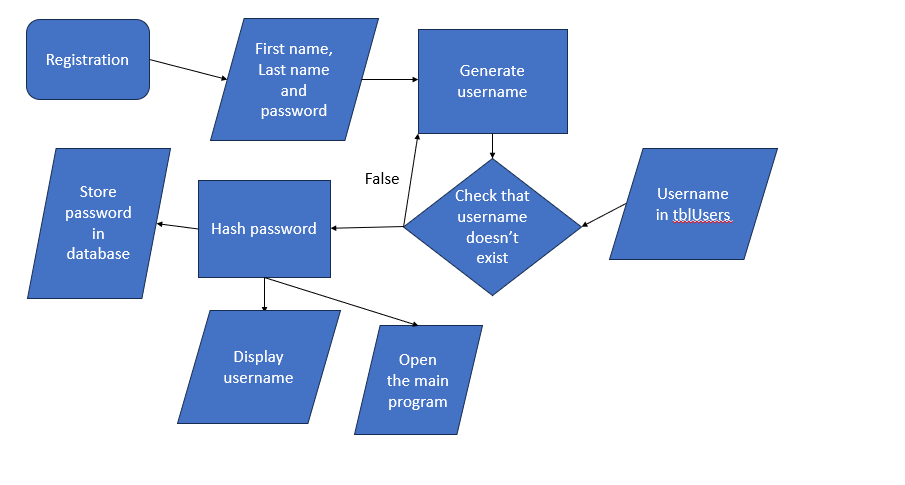
* Advanced programming includes using the data given to the program to be input into the tChart component to display the trends in a personal or company setting and also display how close the user is to their goal.
* The used of a 2-dimensional array to hash a password into an unrecognizable state. This will be done by breaking the password into 2-character part and converting it via ORD commands. It is then checked through the array and an appropriate value is found.

Navigation Diagram

Login system for both type of users



Registration system for both uses



GUI

**Note** - the following will have readable error message if there is a validation error. This is a planned GUI it may be different in the final product.

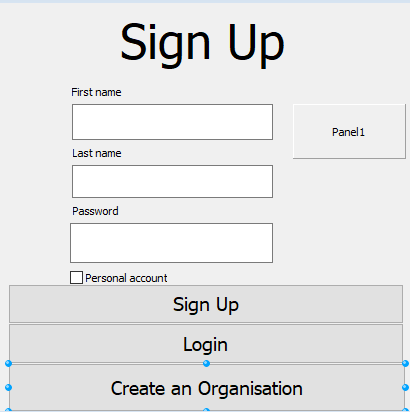
Login form

Graphical user interface

Description automatically generated

|  |  |
| --- | --- |
| Component | Use |
| Tlabel | Used to display useful information to the user |
| Tbutton | Used to run procedure for logging in or to change to the sign up form |
| Timage | Used to display a picture in the background of the form |
| TEdit | Used to enter details such as username and password |

Sign Up form



|  |  |
| --- | --- |
| Component | Use |
| Tlabel | Used to display useful information to the user |
| Timage | Used to display a picture in the background of the form |
| TPanel | Display the generated username to the user |
| TCheckBox | Used to check if the account is a personal account or organizational account |
| TButton | Used to run the sign up procedure or to change the form |
| TEdit | Used to input data like first name, last name and password |

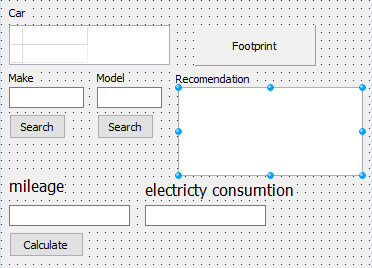
Create organization form

Graphical user interface, application

Description automatically generated

|  |  |
| --- | --- |
| Component | Use |
| Tlabel | Used to display useful information to the user |
| Timage | Used to display a picture in the background of the form |
| TPanel | Display the generated username to the admin |
| TButton | Used to run the create procedure |
| TEdit | Used to the input info such as first name, last name, password and company name |

User form



|  |  |
| --- | --- |
| Component | Use |
| Tlabel | Used to display useful information to the user |
| TDBgrid | Used to display cars in database |
| TEdit | Used to input data to search and to input data to be calculated |
| Tpanel | Used to show the footprint of the user |
| Trichedit | Used to display recommendation to decrease footprint |

IPO table

Input and Validation

Signing up

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Input | Source input | Data type | Format of data | GUI component | Validation |
| First name, last name | Keyboardinput | Strings | Plain text | TEdit | Checks if they only contain letter and the first letter are capital. Will display message saying that the user must check their names. |
| Passwords | Keyboard input | Strings | Plain text | TEdit | Checks if the password is at least 8 character long and has at least one character such as ‘@’ in the password. Will display what reason the password is incorrect such as length lack of unique character. |
| If the account is personal or not | Keyboard input | Boolean | Yes/No | TCheckBox | Checks if the box is checked or not. |
| Sign up | Mouse input |  |  | TButton | Checks if all of the required information has been filled in by the user. Will display a message a message asking you make sure all data is filled in. |

Loggin in

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Input | Source input | Data type | Format of data | GUI component | Validation |
| Username | Keyboard | string | Plain text | TEdit | Will check if the username is in the database. If not in the database then user will be told to check their username or create a new account. |
| Password | Keyboard | string | Plain text | TEdit | Will check if the password is in the databased. If not in the data base then user will be told to check their password. |
| Login | Mouse input |  |  | TButton | Checks if all of the required information has been filled in by the user. Will display a message a message asking you make sure all data is filled in. |

Processing

|  |  |
| --- | --- |
| Name | How will it be done |
| Username generation | The first name and last name of the user will be used to create a unique username for the user. This will be done in the below manner.  input  James  Easson  Output  EassonJ |
| Password encryption | Using the Ord command and a for loop the password will be encrypted so that the password stays safe in the database. It will be like a Caesar cypher. |
| Username/password storage | Data ill be stored into the tblUser database by way of ADO commands. |
| Username during login | The username will be checked if it’s in the tblUser database by way of ADO commands. |
| Password login | Password will run through the encryption algorithm, and this will be compared to appropriate record in the database. |
| Footprint calculation | The user’s footprint will be calculated upon what data they entered such as their cars mileage and their electricity will be compared to a database and a final carbon footprint will be calculated. |
| Car searching | The user entered car type will be searched to see if it is in the tblCars database |
| Recommendation | Will be provided based on the user carbon footprint it will be stored in a database |

Date Output

|  |  |  |
| --- | --- | --- |
| Data to output | Format | Component |
| Carbon footprint of the user | In mass | Tpanel |
| The result of the car search | Data | TDBGrid |