Requirements Analysis

Overview

The objective of this system is to provide a clear and intuitive interface for purchasing and dispensing petrol at a petrol station. The system includes a manager console that allows employees with appropriate permissions to access the transaction history or change the prices of the fuel available.

Considering that this system would be used in a 24/7 environment, the only downtime would be while a manager was changing the price of fuel. Furthermore, the system prioritizes reliability, as any potential failed transaction may result in the loss of a customer.

Visually impaired drivers account for 2-3% of all drivers, and as such, the interface accounts for this by providing large, distinct buttons complemented by contrasting colours.

The age of potential users was also considered. While the average age for drivers is 48 and technological adaptation has advanced significantly, 7-9% of drivers are over 65 years old, and thus the system was designed to be as simple as possible. This was accomplished using buttons, as the chance for error is significantly decreased when users are limited in what they are able to do. The background for these buttons was set to the selection they are meant to represent, as can be seen in the payment type and cash payment interfaces. Furthermore, in the event of a wrong selection, an undo button was implemented, which takes the user back one frame while maintaining the selections made prior to the last. Finally, a cancel transaction button was also implemented, which takes the user back to the main menu and resets all of their selections.

What follows is a breakdown of the individual interface pages:

Main Pages

* Main Menu (index.html)
  + Three buttons: Petrol, Diesel, Manager Console.
    - Petrol redirects to grade select (gradeSelect.html)
    - Diesel redirects to payment (fixedDynamicPayment.html)
    - Manager console
      * Requires login
* Manager Console (homeScreen.html)
  + Two buttons: transactions, set fuel price
    - Displays a history log for all transactions (viewTransactions.html)
    - Set fuel price asks which type of fuel is to be changed (fuelType.html)
      * Petrol asks which grade (typeOfPetrol.html)
        + Selected grade redirects to keypad (changeFuelPrice.html)
      * Diesel redirects to keypad (changeFuelPrice.html)
    - Changing fuel price is done through a keypad, pressing enter saves the changes (successChangedFuelPrice.html)
* Upon fuel being selected, the interface asks is a user wishes to buy a set amount of gas, or a dynamic amount (fixedDynamicPayment.html)
  + Set amount asks is if the user wants to pay by card or cash (paymentInterface.html)
    - If the user wishes to pay by card, they are redirected to the card interface after selecting their amount (creditCard.html)
    - If the user wishes to pay by cash, they are instructed to select amount (cash.html)
      * Asks the user to insert money following selection
  + Dynamic amount
    - Choosing this option redirects to the card interface (creditCard.html), following verification, the user is directed to the fuelling interface, where they can pump any amount up to 120eu. Once the user pumps the amount desired, the card is charged for that amount
* Interface to input money (cash.html)
  + Four buttons: “5eu , 10eu, 20eu, 50eu”
    - Buttons are able to be selected repeatedly, with a total amount displayed at the bottom
    - Buttons should be the bills themselves for usability points.
    - No limit on cash transactions.
    - Card transactions cap at 120eu (real life systems do this to avoid taxes)
  + Undo last increment button
  + Must display how much the selected amount of cash is worth in fuel
* Credit Card System (creditCard.html)
  + Card must be swiped in order to proceed
  + Buttons simulating card being accepted or denied
  + If this page was directed to through the dynamic payment page, allows user to fuel any amount up to 120eu
* Fuelling interface (fuelDisplay.html)
  + Displays real time values for fuel being pumped:
    - Value per litre
    - Current value
    - How much of the transaction has been pumped so far
    - Total sale(max)
* Transaction data (transaction.html)
  + Displays the transaction details
    - Charge card message to simulate that the card has been charged
  + Redirects to main menu after ~10 sec

User Error Handling

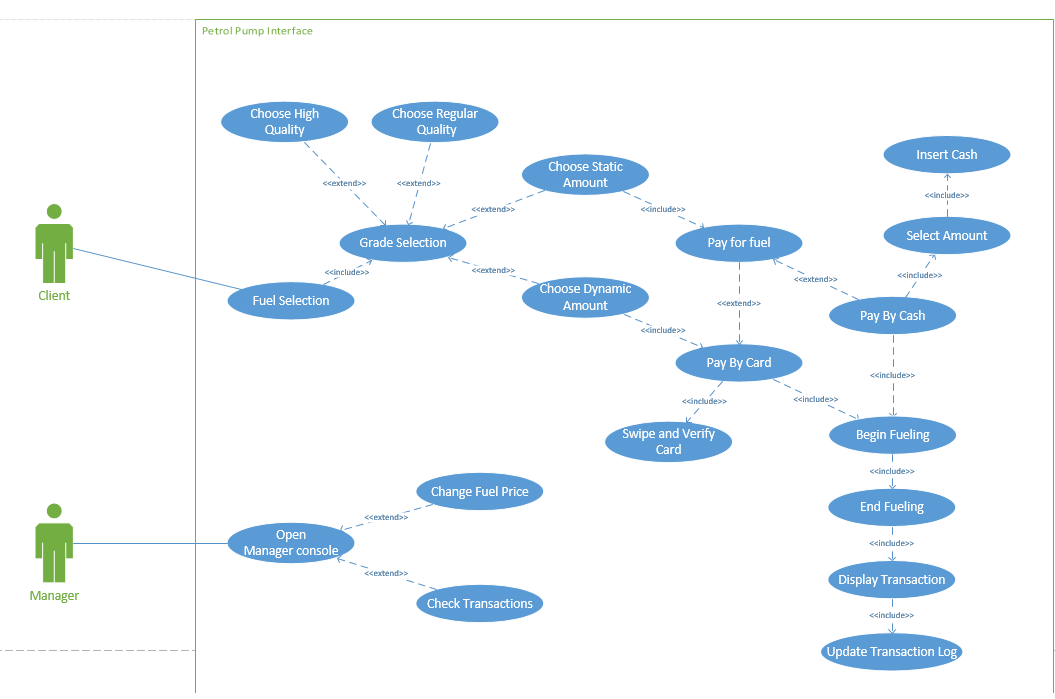
* Go back button
  + Present on every page
  + Redirects the user one page back, clearing any cookies
* Cancel transaction button
  + Redirects the user to the main menu, clearing any cookies

Usability

* Contrasting colours to aid users who are colour-blind
* Large text and buttons to aid users who are visually impaired
* Use of informative backgrounds on buttons to convey information
* Handling of user errors does not reset entire transaction

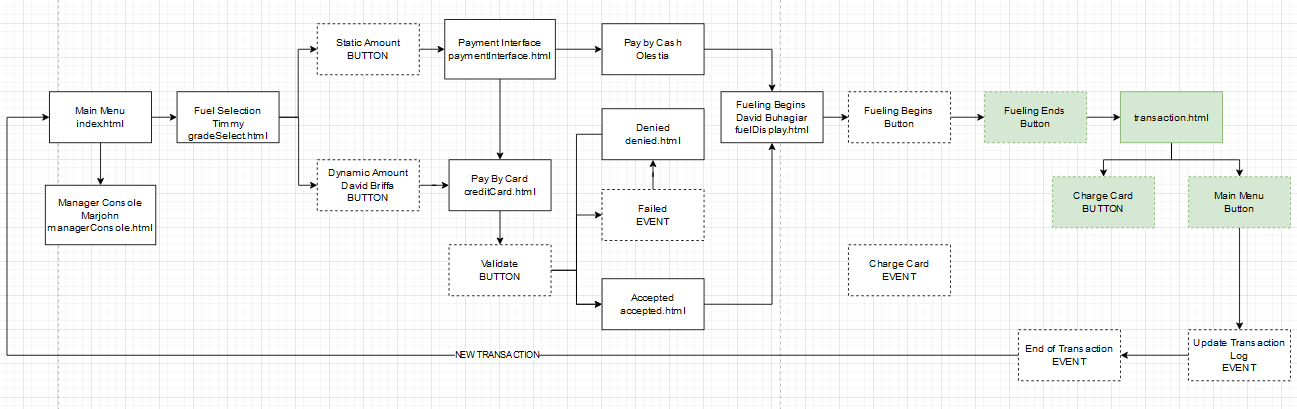
Assumptions

* The software will be used in the EU, and as such will use the Euro currency
* The interface is a touch screen
* The interface is connected to credit card services
* The interface accepts cash through a cash slot
* The interface accepts card through either swiping or contactless

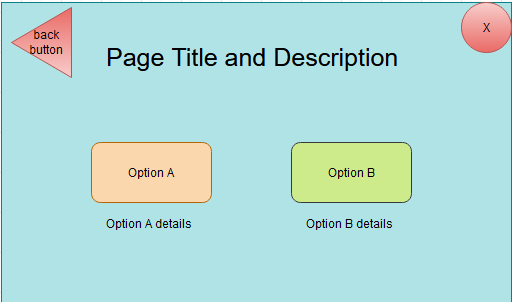
Use Case Diagram

Prototypes

First Prototype

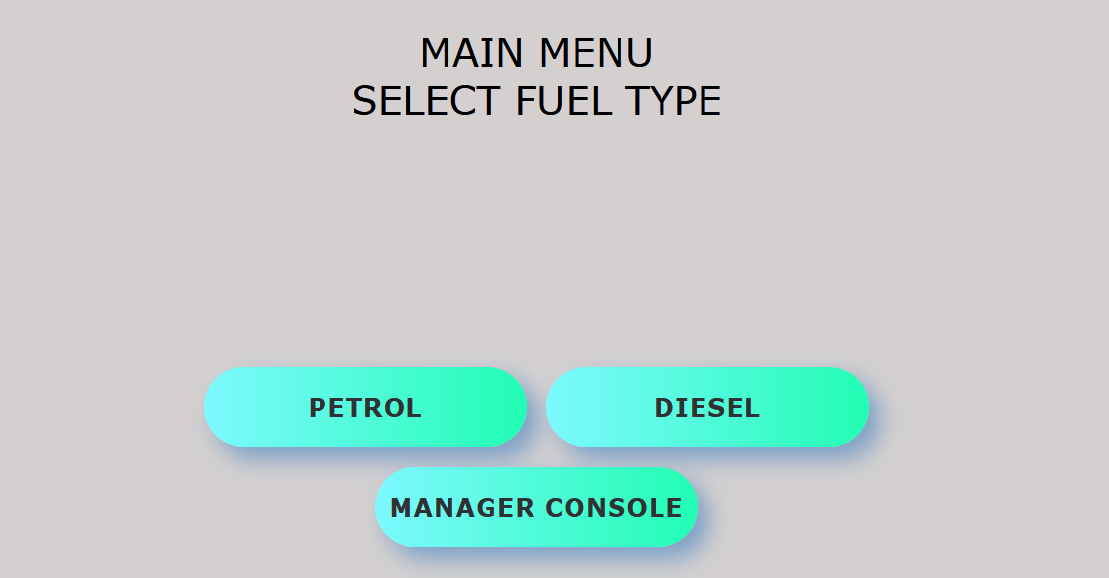


Second Prototype

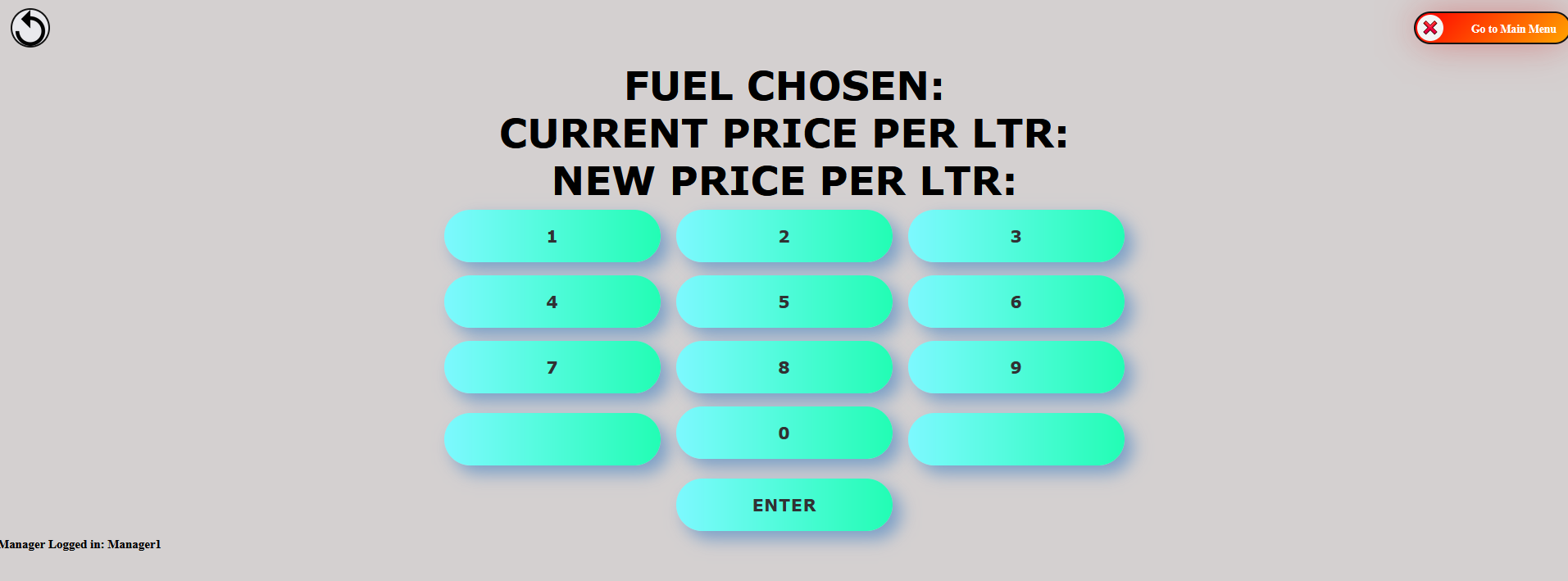


Third Prototype- Implementation

Main menu



Changing fuel price



Selecting payment type



Selecting amount of fuel to buy

