|  |
| --- |
| Robert van der Spoel |
| IT Graad 10 PAT – Fase 1 |
| Circuit Central |
| You may add an image / your own logo (not required) |

Table of Contents

[**Define the task** 2](#_Toc145857759)

[Topic: 2](#_Toc145857760)

[Purpose of program: 2](#_Toc145857761)

[Possible solution: 2](#_Toc145857762)

[Scope: 2](#_Toc145857763)

[**User Stories** 2](#_Toc145857764)

[**Acceptance Test – List FOUR Functions** 2](#_Toc145857765)

[**Data Dictionary** 2](#_Toc145857766)

[User defined function 2](#_Toc145857767)

[**Navigation** 3](#_Toc145857768)

[**GUI Design** 3](#_Toc145857769)

[Tab Sheet 1 (User 1): 3](#_Toc145857770)

[INPUT: 3](#_Toc145857771)

[OUTPUT: 3](#_Toc145857772)

[Tab Sheet 2 (User 2): 3](#_Toc145857773)

[INPUT: 3](#_Toc145857774)

[OUTPUT: 3](#_Toc145857775)

[**IPO – Software Design Tool** 4](#_Toc145857776)

[INPUT VALIDATION: 4](#_Toc145857777)

[PROCESSING: 4](#_Toc145857778)

# **Define the task**

## Topic:

## Purpose of program:

## Possible solution:

## Scope:

# **User Stories**

User 1:

| **WHO** | **WHAT** | **WHY** |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

User 2:

| **WHO** | **WHAT** | **WHY** |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# **Acceptance Test – List FOUR Functions**

* The system
* The system
* The system
* The system

# **Data Dictionary**

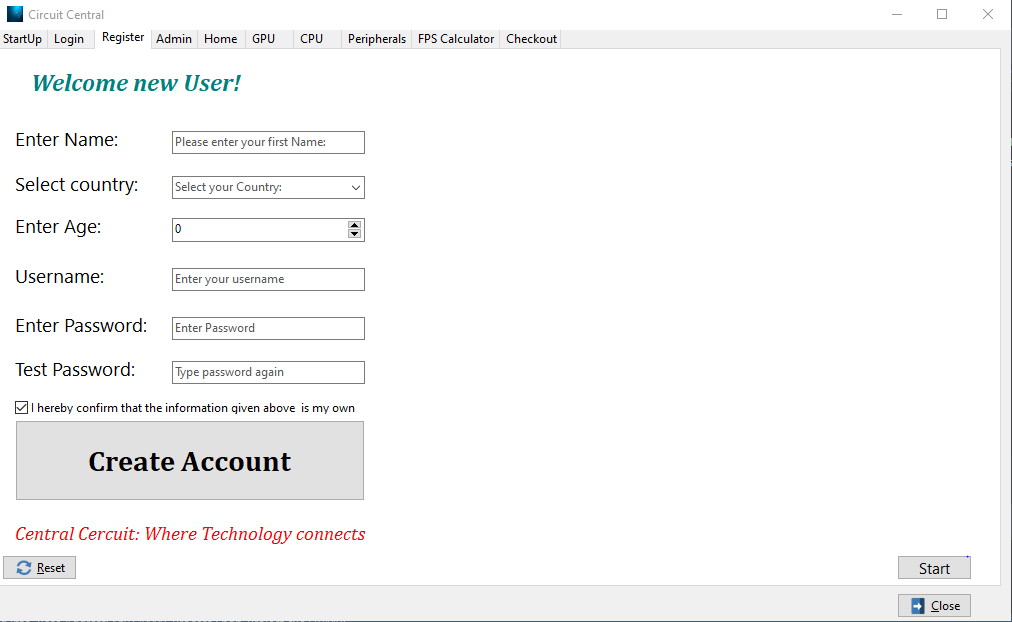
## User defined function

|  |  |
| --- | --- |
| **Signature** | **Explanation** |
|  |  |

# **Navigation**

# **GUI Design**

## Tab Sheet 1 (Register for a new User):



## INPUT:

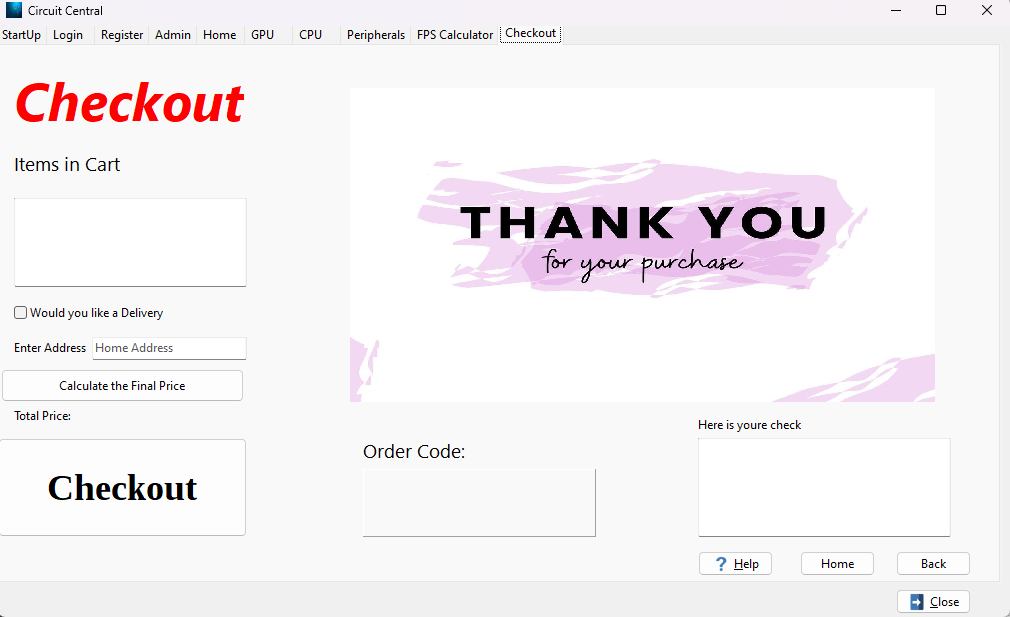
| **Tabsheet 1 (tsRegister , Caption:’Register’)** | | | | |
| --- | --- | --- | --- | --- |
| **Source** | **Variable name & data type** | **Format (size, M/F for gender, yyyy/mm/dd)** | **Object Name** | **Local/Global variable** |
| Edit  Name | sRegisterName  -String | Just Add your name in the edit | edtRegisterName | Global |
| Combobox  Country | iCountry- int  sCountry - str | Select Country  Store the country | cmbCountry | Local |
| Spin edit  Age | iAge-Int  sAge- str | Select or enter age  Store the age | sedAge | Local |
| Edit  Username | sUsernameCreate - string | Create the username | edtUsernameCreate | Global |
| Edit  Password Creation | sPasswordCreate- string | Enter the password that you want | edtPasswordCreate | Local |
| Edit  Check if passwords is the same | sPasswordCheck - string | Re-enter the password that you want | edtPasswordConfirm | Local |
| Checkbox  Confirm that the information thar the user gives is his/her own | bInfoChecked  -Boolean | Check or uncheck the checkbox | chkConfirmDetails | Local |

## OUTPUT:

| **Tabsheet 1 (tsRegister , Caption:’Register’)** | | |
| --- | --- | --- |
| **Output** | **Format (Type, size)** | **Component** |
| Username | Your UserName: <UserName> | Panel |
| Password | Your Password : <Password> | Panel |

## 

## Tab Sheet 2 (Buy GPU’s):



## INPUT:

| **Tabsheet 2 (tsCheckoout caption:’Checkout’)** | | | | |
| --- | --- | --- | --- | --- |
| **Source** | **Variable name & data type** | **Format (size, M/F for gender, yyyy/mm/dd)** | **Object Name** | **Local/Global variable** |
| Checkbox | bDelevery  -Boolean | Select or unselect the checkbox to add funds and delivery to your order or remove it . | chkDelivery | Global |
| Edit  -Collects the street name | sStreetName  -String | Enter your Street Name | edtStreetName | Global |
| Edit  -Collects the house number | iHouseNum  -Integer | Enter your house Number | edtHouseNumber | Global |
| Button  -Calculate Numbers and create an order code | sOrderCode  -String | Generate random numbers and using in and the order Boolean create Order Number for the Customers order | btnCheckout | Global |

## 

## OUTPUT:

| **Tabsheet 2 (tsCheckoout caption:’Checkout’)** | | |
| --- | --- | --- |
| **Output** | **Format (Type, size)** | **Component** |
| Total Price | <Total Price> | label |
| Order Code | <Order Code> | Panel |
| Cheque | <Username>  <Order Details>  <Total Price>  <Order Code> | Rich edit |

# 

# **IPO – Software Design Tool**

## INPUT VALIDATION: For screen One – Register

| **Input** | **Validation technique and description** | **Associated error message** |
| --- | --- | --- |
| Name  -Edit | I will test that there is a name inserted in the Edit | ‘Please Enter your name’ |
| Name  -Edit | I will that the name entered in the edit is 2 characters or longer | ‘Please enter a name with two or more letters’ |
| Country  -Combo box | I will test that a country has been selected in the combo box | ‘Please select a country from the list provided’ |
| Age  -spin edit | I will test that the user has selected a number(There age) in the spin edit | ‘Please select your age’ |
| Age  -Spin edit | I will test that the user is older than 18 years old | ‘You must be at least 18 years old to create an account’ |
| Age  -Spin edit | I will test that the user is younger than 18 years old | ‘You must be younger than 100 years to create an account , please ask a family member for assistance’ |
| Username  -Edit | I will test that there is an username entered in the edit | ‘Please create an username’ |
| Username  -Edit | I will test that the username entered is equal or longer that 5 characters | ‘Username to short , must be at least 5 characters’ |
| Username  -Edit | I will test that the Username entered is equal or shorter than 10 characters | ‘Username to long , must me shorter than 10 characters’ |
| Username  -Edit | I will test that the username entered is only on word | ‘Username can only me one word’ |
| Password  edit | I will test that a password has been entered in the edit | ‘Please enter a Password’ |
| Password  Edit | I will test that the password contains at least 5 characters | ‘Password must contain at least 5 characters’ |
| //Below  Password  Edit | I will test that the password is not longer that 10 characters | ‘Password should be shorter than 10 characters’ |
| Password Test  Edit | I will test that a password has been entered in the edit | ‘Please enter the password again to make sure you know it’ |
| Password Test  Edit | Check that the password entered in the test password edit is the same as the password entered in the create password edit | ‘Passwords do not match’ |
|  |  |  |

## 

## PROCESSING:

|  |  |
| --- | --- |
| **What processing will be done** | **Algorithms, formulas, etc.** |
| **List:** | **Algorithms:** |
| 1Login :  Determine if the user is an admin or a customer or if the user does not exist when the login button is clicked | sUsername = input from user (edit)  sPassword = input from user(Edit)  iTries = 0  sCredentials = sUsername + sPassword  if sCredentials = sCredentialsAdmin  Show admin page  Set to admin page  Show all user Pages  Else  If sCredentials = sCredentialsPresetUserOne  Set to home Page  Make all other user pages available  Else  If If sCredentials = sCredentialsPresetUserCreate  Set to home Page  Make all other user pages available  {Else  iTries + 1  if iTries > 1  button to Register = visible  Show a message: ‘No account? Register Today’  Else  Show error message: ‘Please Check Credentials’} |
| 2Register:  Create a new account for the user and display there login details | sNameRegister = input from user(edit) – Name of User  sCountry = input from user(Combobox)  iAge = input from user(Spin edit)  sRegisterOwnUser= input from user (edit)–User Choose UserName  sPasswordCreate = input from user(edit)- User Choose Password  sPasswordTest = input from user(edit)- User re-enters password  cNamelastLetter = calculate by program  cCountryFirstLetter = calculate by program  cCountryLastLetter = calculated by program  sUserSecondPart = calculated by program  sRegisterUsername = create by program  cNamelastLetter = last letter of sNameRegister  cCountryFirstLetter = first letter of sCountry  cCountryLastLetter = last letter of sCountry  sUserSecondPart = cNamelastLetter + cCountryFirstLetter + cCountryLastLetter + iAge  sRegisterUsername = sRegisterOwnUser + ‘#’ + sUserSecondPart  Show Username  Show Password  Show GO button |
|  |  |