

Screeno Documentation

Developed by : Moath Garaleh
made with ♥ by MoathByte
moathcareerpath@gmail.com

Title Section

❖ Title: Screeno / Visual Basic Project

Introduction:

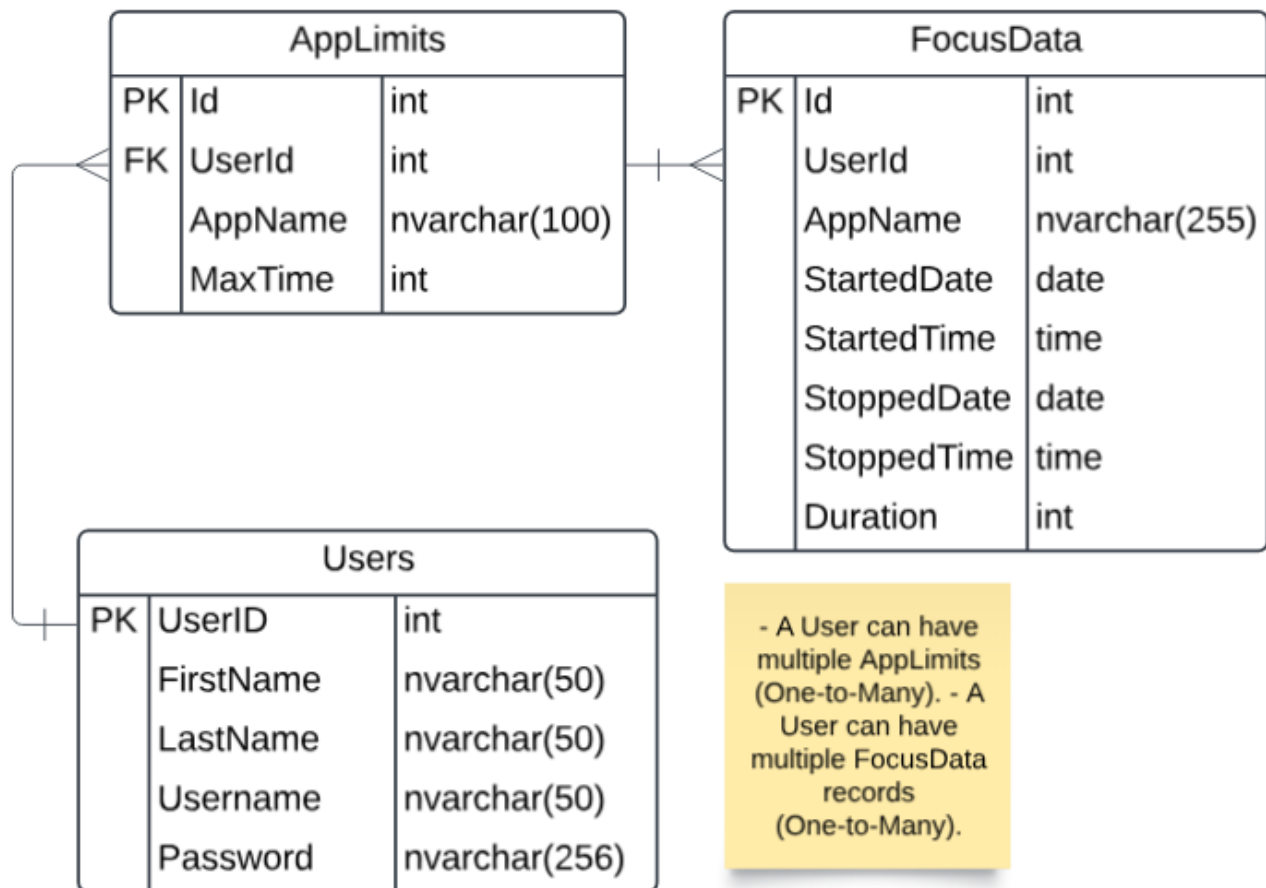
Purpose of the application

- The purpose is to control your usage on your pc, as well as to limit it.

Overview of features

- User friendly UI
- Track usage data & How long you've been using apps on your pc.
- Store usage data in a database
- Limit your application usage.

Database ER Diagram:



Implementation:

We started off by getting inspired by screen time applications on our phones, and decided to go ahead and make a Desktop version of it using Visual Basic.

Our implementation started by figuring out how to capture or record user interaction with other applications, to record that usage. After doing some research and stumbling upon different resources we stumbled upon the Windows API , where it grants us special functions we could use to determine which application is in the Foreground.

Login.vb:

Login vb includes a normal search and compare procedure from the Users table in our database.

For the implementation part of things we decided to be a bit different and hash the passwords before adding them to the database to practice security procedures when developing an application.

```
Private Function HashPassword(password As String) As String
    Using hasher As SHA256 = SHA256.Create()
        Dim bytes As Byte() = hasher.ComputeHash(Encoding.UTF8.GetBytes(password))
        Return Convert.ToBase64String(bytes)
    End Using
End Function
```

Figure 1.2

We used SHA256 to hash and secure the user's password before it is stored in the password field on our database. The HashPassword function is called in the login logic.

Register.vb

Retrieves user information and details from text fields and then adds them to the database, with a button to go back to the login form.

Implementation (Continued)

The **Dashboard.vb** file contains numerous functions, of which are necessary to capture what application is in the foreground and start timingit.

```

Private Sub LogActiveApplication()
    Dim hWnd As IntPtr = GetForegroundWindow()
    Dim processId As UInteger
    GetWindowThreadProcessId(hWnd, processId)

    Dim processName As String = GetProcessName(processId)

    ' Update duration for the current app periodically
    Dim duration As TimeSpan = DateTime.Now - focusStartTime
    If processName = currentApp Then
        ' Save incremental duration
        SaveAppUsageToDatabase(currentApp, focusStartTime, DateTime.Now, duration)
        focusStartTime = DateTime.Now
    Else
        ' Switch to a new app
        If currentApp <> "" Then
            SaveAppUsageToDatabase(currentApp, focusStartTime, DateTime.Now, duration)
        End If
        currentApp = processName
        focusStartTime = DateTime.Now
    End If
End Sub

```

Figure I.1

As shown above in figure I.1 LogActiveApplication starts by calling the Windows API function GetForegroundWindow() and gets the process of that Window. Then saves when it starts so it could save it to the database later on.

The function responsible for getting such data is **LogActiveApplication()**

Dynamic Implementation:

Since our capture data is really important to display to the user, our approach was to dynamically push the data onto a flyoutpanel.

We created a ShadowPanel which was using the Guna2ShadowPanel via code That creates a panel and then adjusts it's properties accordingly

We also used dynamic labels and dynamic buttons.

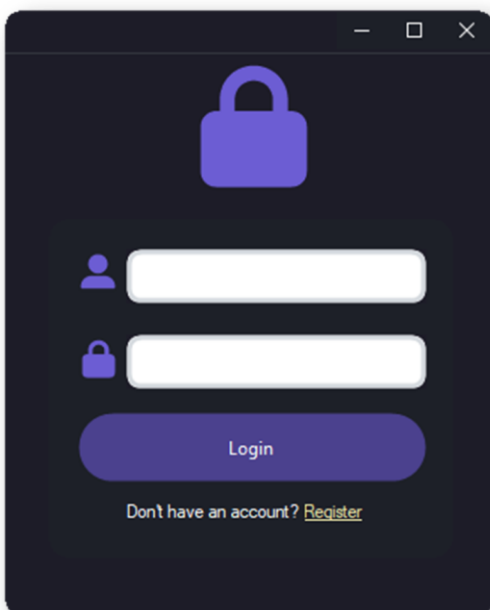
Our functions and a brief explanation on what they do:

FocusTimer_Tick: Updates timer display every second.

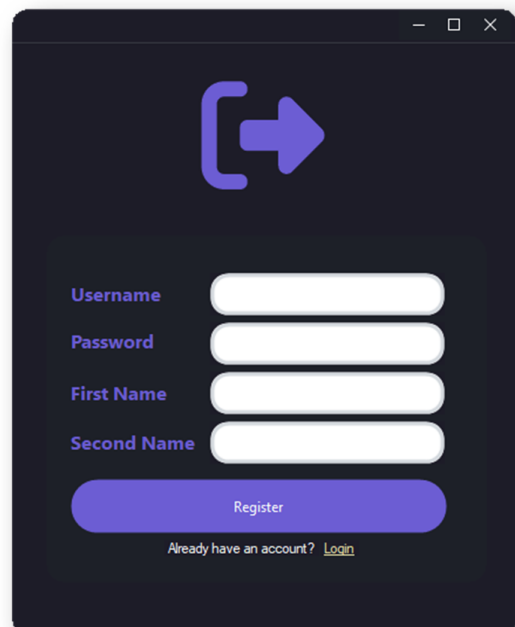
btnFocusMode_CheckedChanged: Starts/stops timer based on selected focus mode.

CheckAppUsageLimits: Checks app usage against set limits.
TerminateApplication: Terminates a running application.
Dashboard_Load: Initializes form and starts timers.
Timer1_Tick: Logs active application and updates app cards.
Timer2_Tick: Checks app usage limits and updates totals.
LogActiveApplication: Logs active app usage to the database.
UpdateTotalUsage: Displays total usage time for the user.
SaveAppUsageToDatabase: Saves app usage data to the database.
GetProcessName: Retrieves process name by process ID.
UpdateAppCards: Updates UI cards with app usage data.
UpdateGraph: Updates the usage graph with app data.
GetAppDurations: Retrieves total durations of apps from database.
GetDetailedAppUsage: Retrieves detailed app usage history.
Guna2ControlBox6_Click: Closes the dashboard and login form.
btnLogout_Click: Logs out the user and shows login form.
btnSetting_Click_1: Opens settings form for the logged-in user.
ClearUsageButton_Click: Clears user app usage data.
ClearLimitButton_Click: Clears user app limits data.
btnReport_Click: Opens email report prompt for issues.

Our Login and Register GUI's:

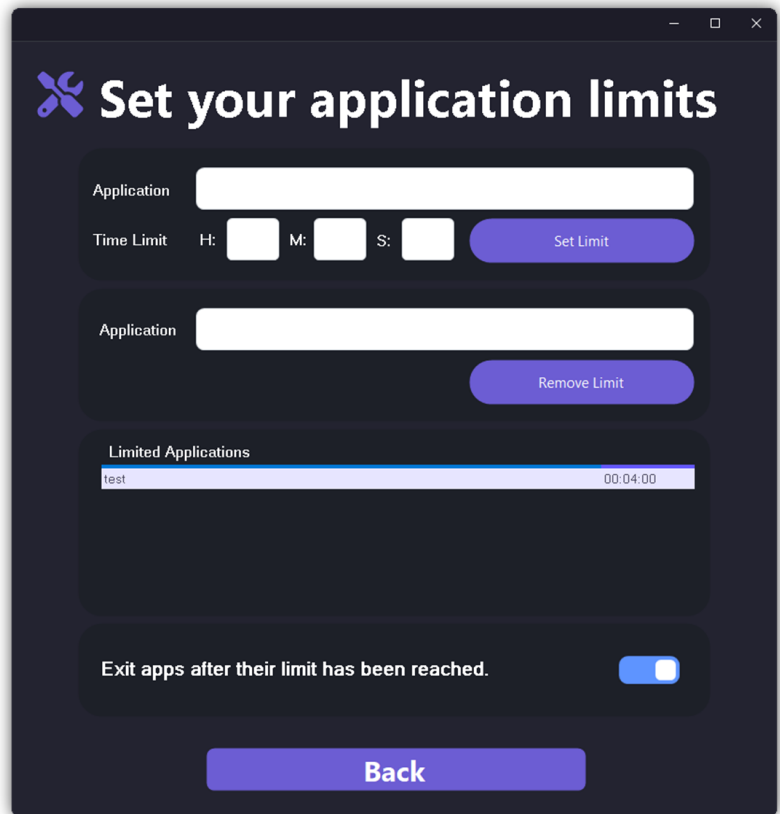


The Login GUI is a dark-themed window with a purple padlock icon at the top. It features two white input fields: the first is preceded by a purple user icon, and the second by a purple padlock icon. Below the fields is a purple 'Login' button. At the bottom, it says 'Don't have an account? [Register](#)'.



The Register GUI is a dark-themed window with a purple icon of a bracket and an arrow at the top. It features four white input fields labeled 'Username', 'Password', 'First Name', and 'Second Name'. Below the fields is a purple 'Register' button. At the bottom, it says 'Already have an account? [Login](#)'.

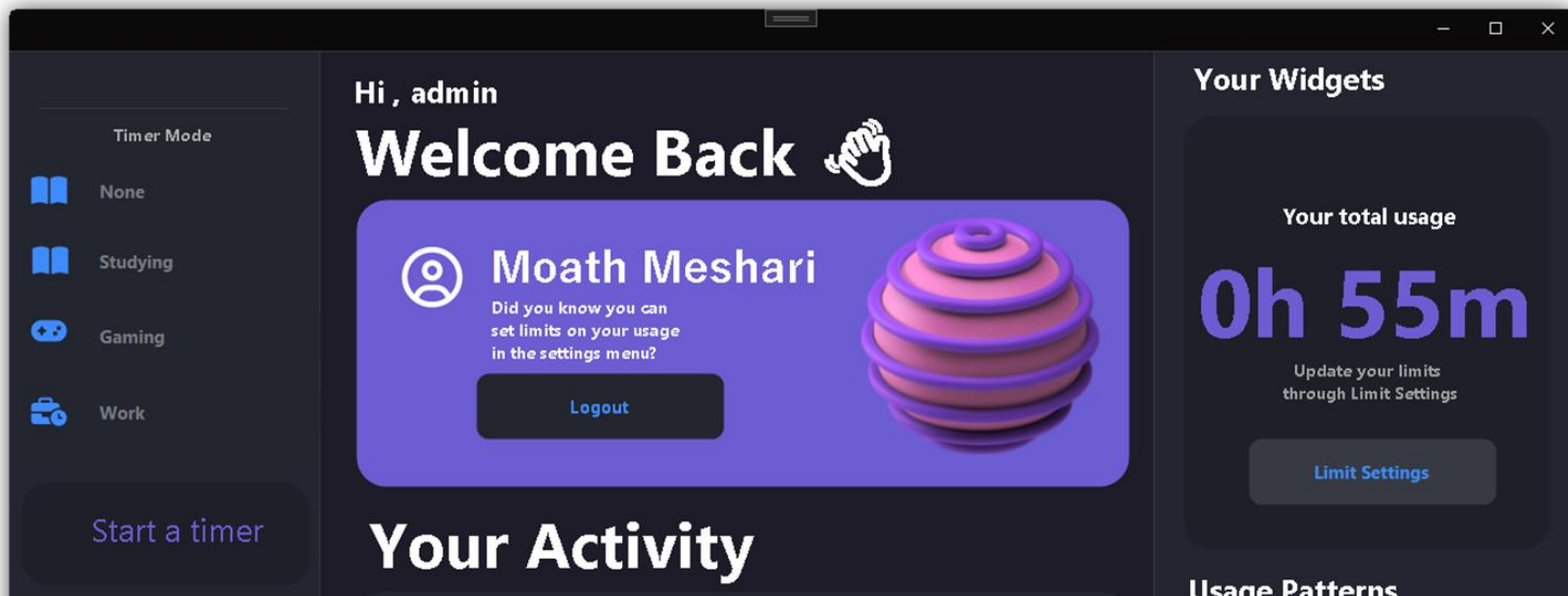
The Main Dashboard:



A dark-themed dialog box titled "Set your application limits" with a wrench icon. It contains two sections for setting limits. The first section has an "Application" text input, a "Time Limit" section with "H:", "M:", and "S:" inputs, and a "Set Limit" button. The second section has an "Application" text input and a "Remove Limit" button. Below these is a "Limited Applications" table with one row: "test" and "00:04:00". At the bottom, there is a toggle switch for "Exit apps after their limit has been reached." and a "Back" button.

Limited Applications	
test	00:04:00

The Main Dashboard:



The main dashboard is a dark-themed interface. On the left is a sidebar with "Timer Mode" and a list of categories: None, Studying, Gaming, and Work. The main content area has a greeting "Hi, admin" and "Welcome Back" with a hand icon. Below this is a profile card for "Moath Meshari" with a "Logout" button and a 3D sphere graphic. A "Start a timer" button is at the bottom left. On the right, a "Your Widgets" section shows "Your total usage" as "0h 55m" with a "Limit Settings" button. The bottom right corner shows "Usage Patterns".

Hi, admin

Welcome Back

Moath Meshari
Did you know you can set limits on your usage in the settings menu?
[Logout](#)

Your Activity

Your Widgets

Your total usage
0h 55m
Update your limits through Limit Settings
[Limit Settings](#)

Usage Patterns