**CSC 2920 Software Engineering Report**

**11/19/2019**

**Scheduling Application**

**Group #4**

|  |
| --- |
| Gabriel Long |
| Daniel Kakwata |
| Luke Fleming |
| Gary Locklear |
| Logan Craig |
| Gartrel Barnes |

**Table of Contents**

**1.Introducton…………………………………………………………………….1**

**1.1.** Project overview……………………………………………………………..1

**1.2.** Problem Statement…………………………………………………………..1

**1.3.** Project deliverables…………………………………………………………..1

**2.Project Management Plan……………………………………………………2**

**2.1.** Project organization…………………………………………………………2

**2.2.** Process Model ………………………………………………………………3

**2.3.** Schedule…………………………………………………………………….4

**3. Requirement Specifications …………………………………………………5**

**3.1** Overall requirement specifications ………………………………………….5

**3.2** Use Cases…………………………………………………………………..6-8

**3.3** Functional requirements…………………………………………………….9

**3.4** Non-functional requirements………………………………………………..9

**4.** **Design………………………………………………………………………...10**

**4.1** GUI (Graphical User Interface) design……………………………………10-12

**4.2** Sequence Diagram…………………………………………………………..13

**4.3** Class Diagram………………………………………………………………14

**5. Implementation……………………………………………………………..15**

**6. Test Plan…………………………………………………………………….16**

**6.1** Techniques used for test generation………………………………………..16

**6.2** Test Plan…………………………………………………………………17-18

**Appendix……………………………………………………………………19-30**

**Introduction**

* 1. **Project overview**

This document entails the details of a visual studio program that will be used to schedule and view appointments. The goal is to have a visual program that allows a user to set and view appointments for ease of use. The user will be able to add themselves to an appointment list as well as view and remove themselves at their discretion. There will also be the functionality of a login/registry further down the line in development.

* 1. **Problem Statement**

With the advent of the digital age has brought on the need to modernize and improve every aspect of business, one such improvement needed is the way in which we set our schedules. The objective is to design a program that has a registry of names that can be accessed by clients of a company, so that they may set their own schedules and appointments without the need of going through a third party or office to set the appointments

**1.3 Problem Deliverables**

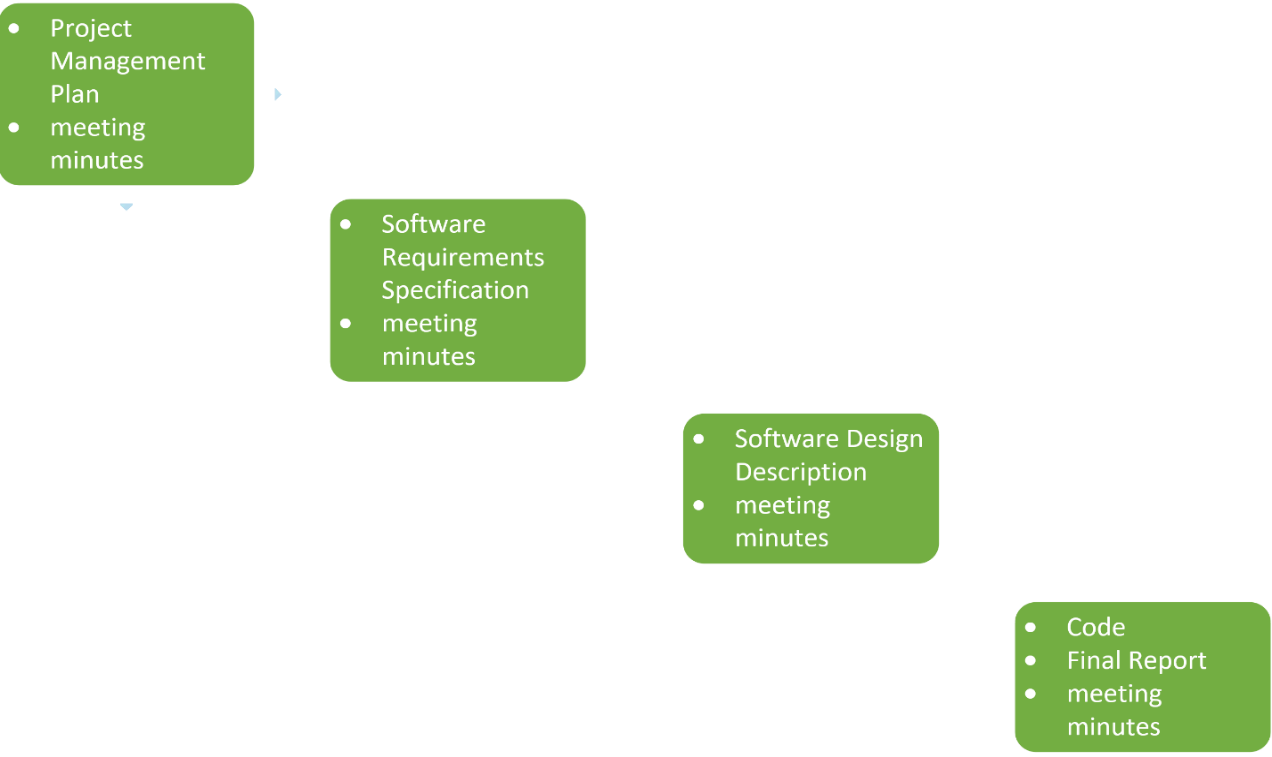
|  |  |  |
| --- | --- | --- |
| **Deliverable Number** | **Deliverable Name** | **Due Date** |
| **1.** | Project Management Plan  (30 minutes) | 10/28/2019 |
| **2.** | Software Requirements Specification | 11/04/2019 |
| **3.** | Software Design Description | 11/11/2019 |
| **4.** | Final Report | 11/19/2019 |

**2. Project management plan**

**2.1 project organization**

|  |  |  |  |
| --- | --- | --- | --- |
| Group Member | Role | Skills | Training Needed |
| Gabriel  Long | Team Leader | Communications, Programming: Java, Visual Studio | Additional Programming |
| Daniel  Kakwata | Implementor, Configuration Manager | Programming: Java, | Object-oriented databases  UML Modeling |
| Luke  Fleming | Implementor, Tester | Programming: Java, Python, Javascript | UML Modeling, UI |
| Gary  Locklear | Implementor, Tester | Programming: Java | Additional Programming, UML Modeling |
| Logan  Craig | Implementor, | Programming: Java  Linus, Mars | Visual Basic |
| Gartrel Barnes | Implementor | Programming: Java, C++  Design | Additional Programming |

**2.2 SW Development Process Model**



Linear sequential model

Brief description

We will go about making this project by following the steps laid out for us which would mean the first stage would be project management. The second step would consist of software requirements. The third will include a software design description. And the fourth and final step will include both the code and the final report on the project. All Steps will include meeting minutes.

**2.3 Scheduling**

Program Name: Scheduling Application



**3.Requirement Specifications**

**3.1Overall Requirement Specification**

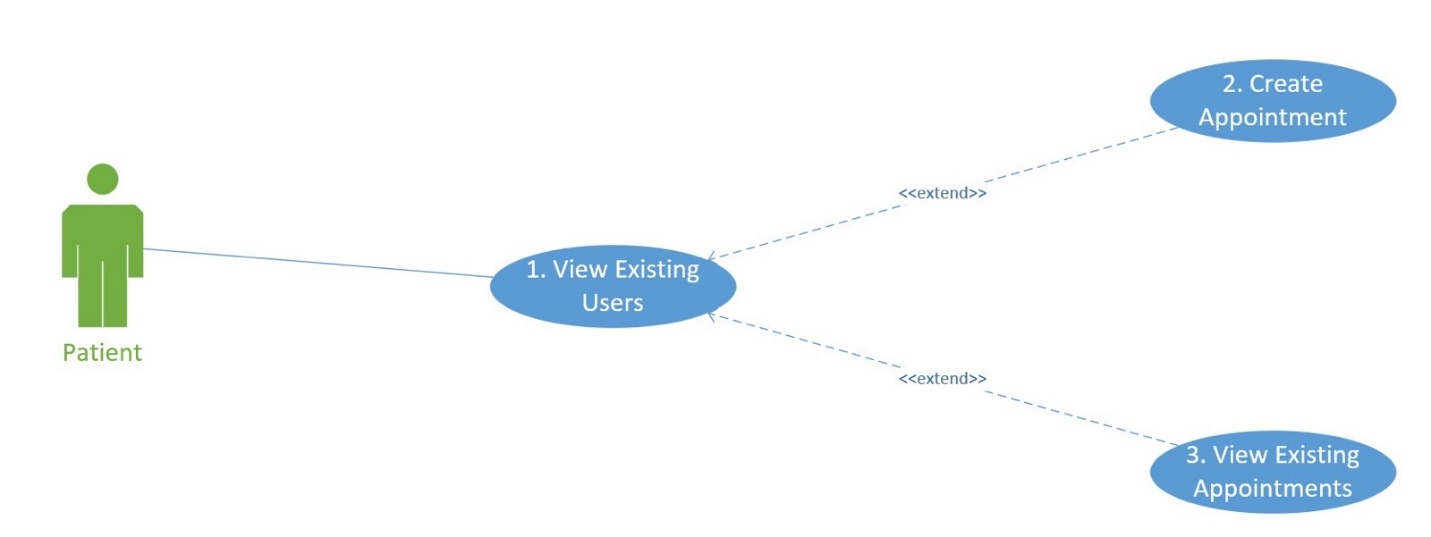
*Intended Audience and use*

The intended audience for this program is major businesses that require face to face meetings regularly, its use is intended to be for making meeting times easier to set up without the need of a secretary.

*Product scope*

For the Business Application, our group decided to approach with 3 use cases, to keep the overall work simplified within the project. Luke and Daniel worked on the use case diagram and description and released it first hand, so that other group members could use it to complete their respective parts. We are not planning on lot of security (i.e. logging into a user account versus an admin account) within the version of the app we are developing, due to time constraints. What we are planning on, however, is to build the program so that everyone will have the option to create and view appointments, by simply choosing a user within our database within the app. This way, we only have one actor, and thus only three use cases to actually implement during the course of the project.

**3.2: Use Case Diagram and Use Case Tables:**

****

View existing User

|  |  |
| --- | --- |
| **Use case name** | View existing users |
| **Participating Actors** | Patient |
| **Entry condition** | None |
| **Exit condition** | None |
| **Flow of the event** | 1. Enter the home screen 2. Request to see view existing users 3. View existing users 4. Option to create a new appointment or view existing appointment. |

Create Appointment

|  |  |
| --- | --- |
| **Use case name** | Create Appointment |
| **Participating Actors** | Patient |
| **Entry condition** | View Existing Appointments |
| **Exit condition** | None |
| **Flow of the event** | 1. Enter Home Screen 2. Request to view existing users 3. View Existing Users 4. Request to Create appointment for client selected 5. Create Appointment 6. Add new appointment to list of Existing appointments |

View Existing Appointments

|  |  |
| --- | --- |
| **Use case name** | View existing Appointments |
| **Participating Actors** | Clients and Patients |
| **Entry condition** | View Existing Users |
| **Exit condition** | None |
| **Flow of the event** | 1. Enter the home screen 2. Request to see view existing users 3. View existing users 4. Request to view existing appointments 5. View Existing Appointments |

**3.3 Functional Requirements**

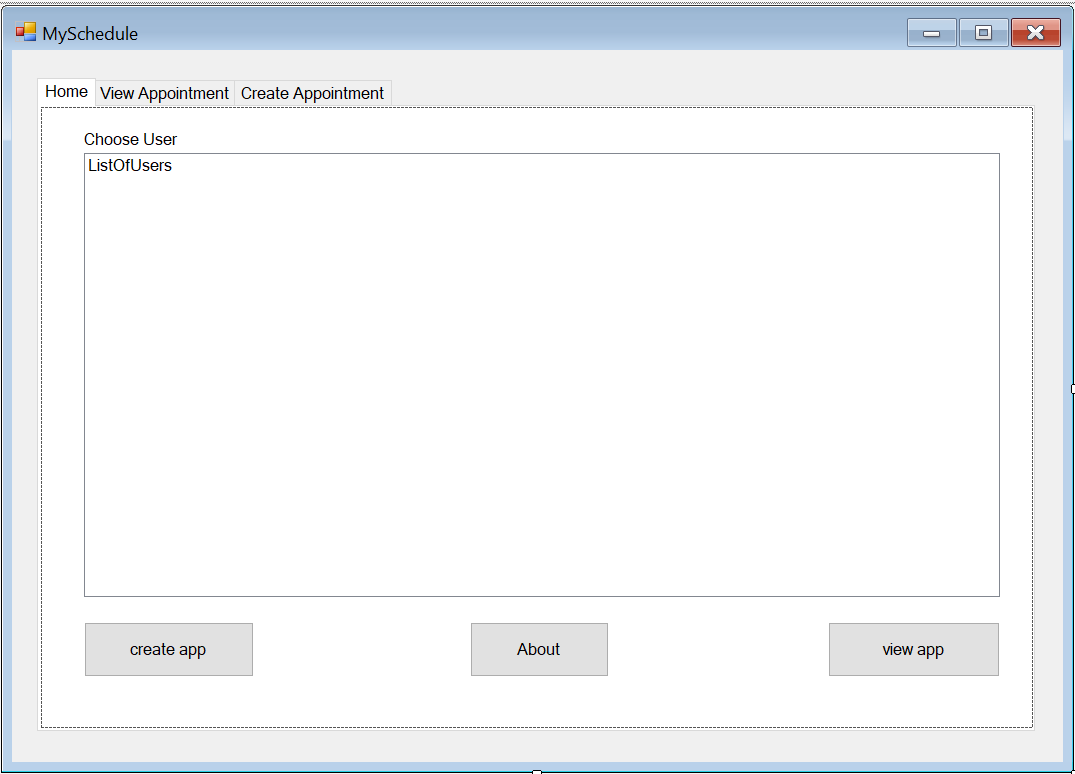
* A login textbox
* A textbox that is used to enter and view the appointment list
* Registry of users
* The registry of users that can be accessed and viewed by users
* Appointment creation function
* function will take the user selected from the registry and create an appointment and add the information to the registry of appointments.

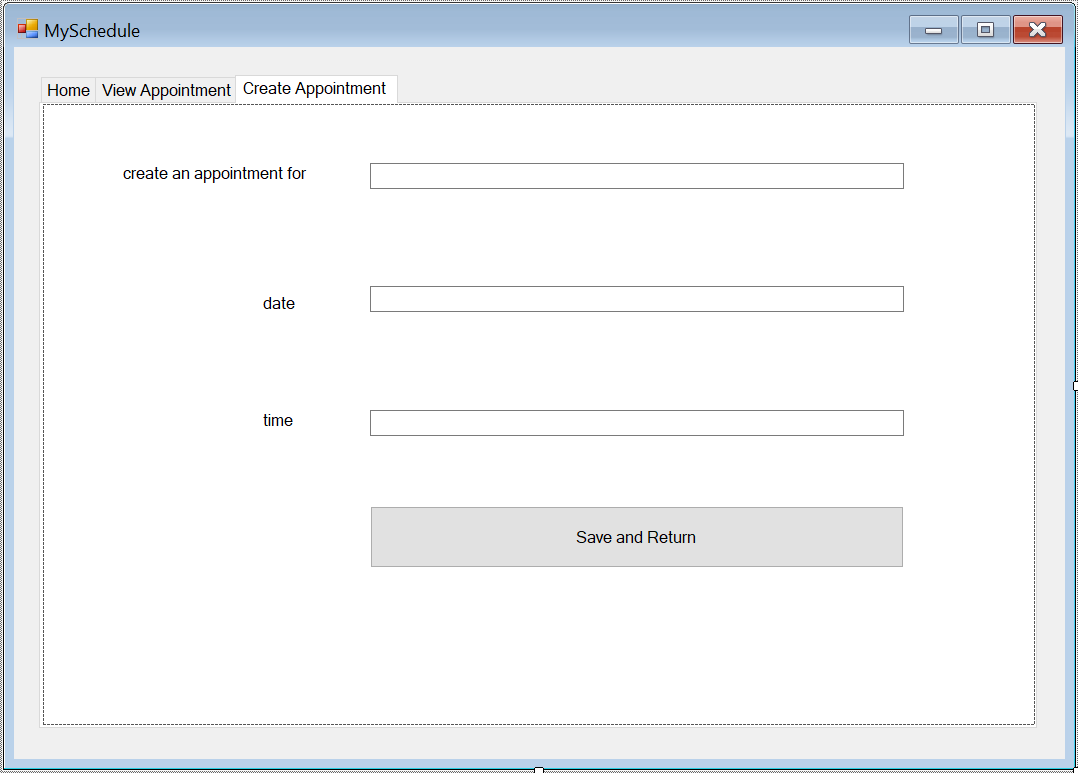
**3.4 Non- Functional Requirements**

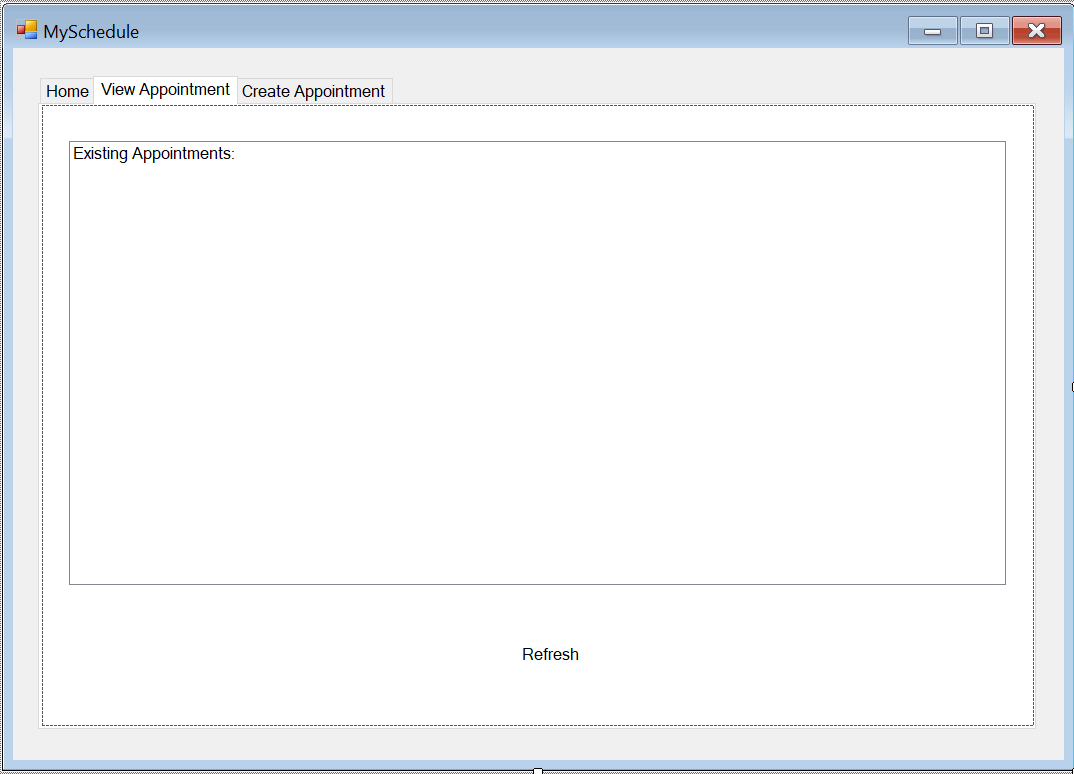
* Loading Times
* The program must run in 1 sec
* Usability
* Must be accessible via apple Itunes, android store

**4.Design**

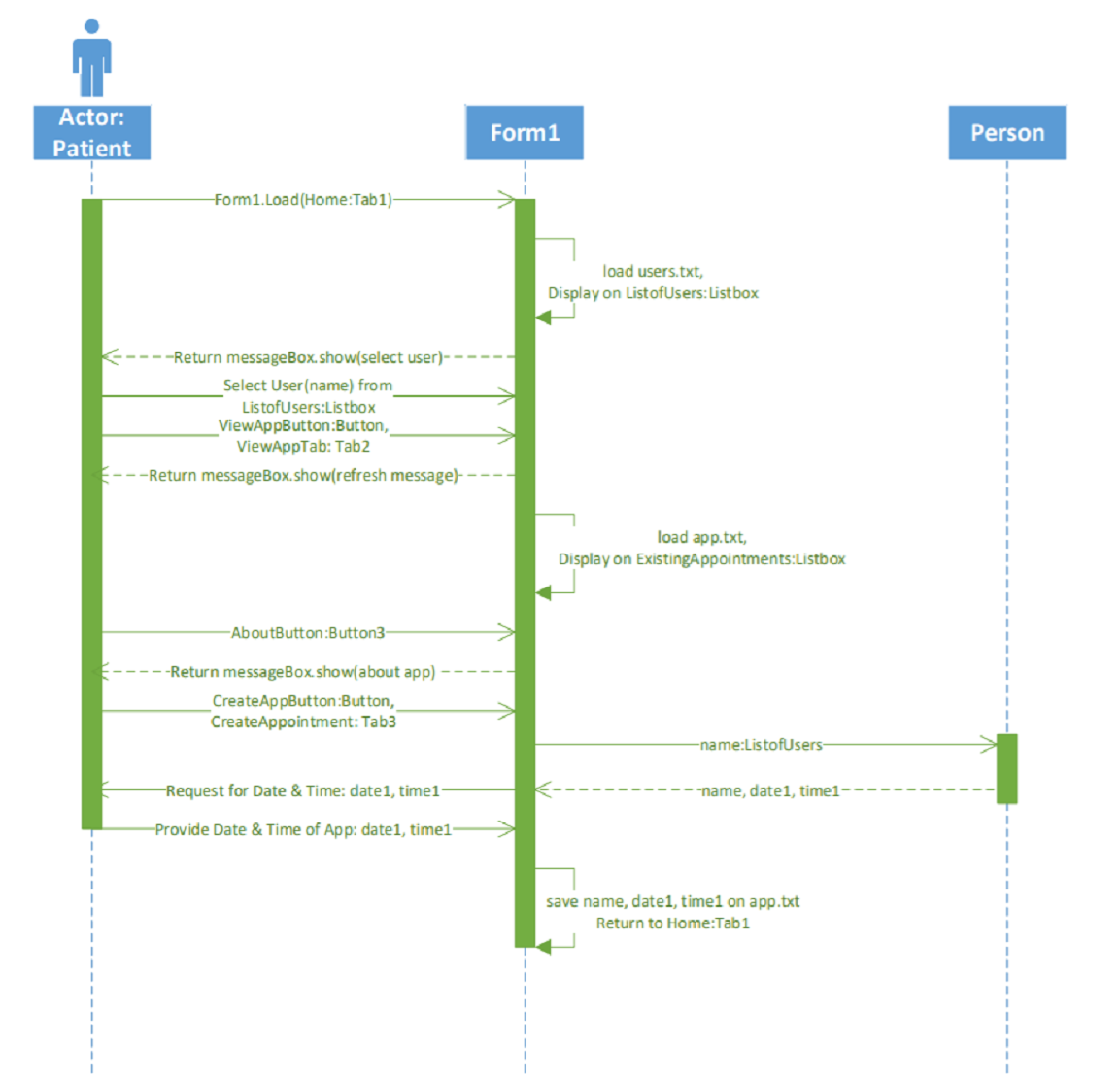
4.1 Graphical User interface







4.1 Sequence diagram



4.3 Class Diagram

****

**5.Implementation**

Brief Description of code:

For our application, upon opening it, the actor will have a listbox of users to choose from (retrieved from a text file). They will have three options to select once they have selected a user from the listbox; create an appointment, about, and view appointments. On the create appointment tab, the actor will then fill out a small form with the date and time of the appointment. When the actor selects Save and Return, the application will save the data to a text file. Then, after returning to the home tab, the actor should be able to select view appointment, and (after hitting refresh outside of the listbox) should be able to see their appointment. If the actor selects the about button, a message box will show with a short description about the application.

Classes:

The person class contains variables for the date and time of the appointment, along with the name passed from the Form1 classes database (users.txt). The Form1 class will get the variables name, date1,and time1 from the person class, in order to successfully create the appointment. The Form1 class contains everything else pertaining to operation of the app, the user database (users.txt), and the appointment database (app.txt).

Note: Person Class is only active during the create appointment process. Form1 Class is active as long as the program is open.

\*Full code located in appendix

**6.Test plan**

6.1. Black box testing

For this project, we used Black box testing. We just focus on inputs and output of the software system without bothering about internal knowledge of the software program. We described all possible inputs, all expected system outcome and the result.

6.2 Test plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case ID: 01 Test name: Home screen  System: appointment App Executed by: Daniel Kakwata  Short description: Program display a message to select a user before pressing a button. | | | | |
| Step | Input | Expected System Outcome | Result | fix |
| 1 | Click ok | The system displays a list of existing users | Print success |  |
| 2 | choose a user | By choosing a user, the system gives options to create an appointment or view appointment if there is any | success |  |
| 3 | Click about | System displays all information about the application | Print success |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case ID: 02 Test name: view appointment  System: appointment App Executed by: Daniel Kakwata  Short description: Program will display a list of all appointment created | | | | |
| Step | Input | Expected System Outcome | Result | fix |
| 1 | Click view appointment on the bottom line | The system displays a notice for the user to click under the box to view existing appointment | Print success |  |
| 2 | Click ok | Existing appointment page will come out blank | Print success |  |
| 3 | Click under the box | System will display a list of all existing appointment | Print success |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case ID: 03 Test name: create appointment  System: appointment App Executed by: Daniel Kakwata  Short description: Program gives option to create a new appointment or modify an existing appointment | | | | |
| Step | Input | Expected System Outcome | Result | fix |
| 1 | Click create appointment | The system displays a screen to create an appointment | Print success |  |
| 2 | Enter James, 3/12/2019, and 3PM | System keeps the information visible on the screen | success |  |
| 3 | Click save and return | System notify the user that he will be redirected to the view appointment tab | Print success |  |
| 4 | Click ok | System displays a list of existing appointment | Print success |  |
| 5 | Click under the box | System will display all appointment | Print success |  |

**Appendix**

**<Global.Microsoft.VisualBasic.CompilerServices.DesignerGenerated()>**

**Partial Class Form1**

**Inherits System.Windows.Forms.Form**

**'Form overrides dispose to clean up the component list.**

**<System.Diagnostics.DebuggerNonUserCode()>**

**Protected Overrides Sub Dispose(ByVal disposing As Boolean)**

**Try**

**If disposing AndAlso components IsNot Nothing Then**

**components.Dispose()**

**End If**

**Finally**

**MyBase.Dispose(disposing)**

**End Try**

**End Sub**

**'Required by the Windows Form Designer**

**Private components As System.ComponentModel.IContainer**

**'NOTE: The following procedure is required by the Windows Form Designer**

**'It can be modified using the Windows Form Designer.**

**'Do not modify it using the code editor.**

**<System.Diagnostics.DebuggerStepThrough()>**

**Private Sub InitializeComponent()**

**Me.tabCreateAppointment = New System.Windows.Forms.TabPage()**

**Me.TextBox6 = New System.Windows.Forms.TextBox()**

**Me.TextBox3 = New System.Windows.Forms.TextBox()**

**Me.TextBox2 = New System.Windows.Forms.TextBox()**

**Me.ReturnButton = New System.Windows.Forms.Button()**

**Me.timeLabel = New System.Windows.Forms.Label()**

**Me.dateLabel = New System.Windows.Forms.Label()**

**Me.CreateAnAppLabel = New System.Windows.Forms.Label()**

**Me.tabViewAppointment = New System.Windows.Forms.TabPage()**

**Me.CreateAppointmentButton = New System.Windows.Forms.Button()**

**Me.ListBox1 = New System.Windows.Forms.ListBox()**

**Me.tabHome = New System.Windows.Forms.TabPage()**

**Me.Button3 = New System.Windows.Forms.Button()**

**Me.Button2 = New System.Windows.Forms.Button()**

**Me.Button1 = New System.Windows.Forms.Button()**

**Me.AboutButton = New System.Windows.Forms.Button()**

**Me.ViewAppointmentButton = New System.Windows.Forms.Button()**

**Me.CreateUserSend = New System.Windows.Forms.Button()**

**Me.ListOfUsers = New System.Windows.Forms.ListBox()**

**Me.Label1 = New System.Windows.Forms.Label()**

**Me.TabControl1 = New System.Windows.Forms.TabControl()**

**Me.Label2 = New System.Windows.Forms.Label()**

**Me.tabCreateAppointment.SuspendLayout()**

**Me.tabViewAppointment.SuspendLayout()**

**Me.tabHome.SuspendLayout()**

**Me.TabControl1.SuspendLayout()**

**Me.SuspendLayout()**

**'**

**'tabCreateAppointment**

**'**

**Me.tabCreateAppointment.Controls.Add(Me.TextBox6)**

**Me.tabCreateAppointment.Controls.Add(Me.TextBox3)**

**Me.tabCreateAppointment.Controls.Add(Me.TextBox2)**

**Me.tabCreateAppointment.Controls.Add(Me.ReturnButton)**

**Me.tabCreateAppointment.Controls.Add(Me.timeLabel)**

**Me.tabCreateAppointment.Controls.Add(Me.dateLabel)**

**Me.tabCreateAppointment.Controls.Add(Me.CreateAnAppLabel)**

**Me.tabCreateAppointment.Location = New System.Drawing.Point(4, 29)**

**Me.tabCreateAppointment.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.tabCreateAppointment.Name = "tabCreateAppointment"**

**Me.tabCreateAppointment.Padding = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.tabCreateAppointment.Size = New System.Drawing.Size(992, 621)**

**Me.tabCreateAppointment.TabIndex = 3**

**Me.tabCreateAppointment.Text = "Create Appointment "**

**Me.tabCreateAppointment.UseVisualStyleBackColor = True**

**'**

**'TextBox6**

**'**

**Me.TextBox6.Location = New System.Drawing.Point(327, 59)**

**Me.TextBox6.Margin = New System.Windows.Forms.Padding(3, 4, 3, 4)**

**Me.TextBox6.Name = "TextBox6"**

**Me.TextBox6.Size = New System.Drawing.Size(534, 26)**

**Me.TextBox6.TabIndex = 7**

**'**

**'TextBox3**

**'**

**Me.TextBox3.Location = New System.Drawing.Point(327, 306)**

**Me.TextBox3.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.TextBox3.Name = "TextBox3"**

**Me.TextBox3.Size = New System.Drawing.Size(534, 26)**

**Me.TextBox3.TabIndex = 5**

**'**

**'TextBox2**

**'**

**Me.TextBox2.Location = New System.Drawing.Point(327, 182)**

**Me.TextBox2.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.TextBox2.Name = "TextBox2"**

**Me.TextBox2.Size = New System.Drawing.Size(534, 26)**

**Me.TextBox2.TabIndex = 4**

**'**

**'ReturnButton**

**'**

**Me.ReturnButton.Location = New System.Drawing.Point(327, 402)**

**Me.ReturnButton.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.ReturnButton.Name = "ReturnButton"**

**Me.ReturnButton.Size = New System.Drawing.Size(534, 62)**

**Me.ReturnButton.TabIndex = 6**

**Me.ReturnButton.Text = "Save and Return"**

**Me.ReturnButton.UseVisualStyleBackColor = True**

**'**

**'timeLabel**

**'**

**Me.timeLabel.AutoSize = True**

**Me.timeLabel.Location = New System.Drawing.Point(216, 306)**

**Me.timeLabel.Name = "timeLabel"**

**Me.timeLabel.Size = New System.Drawing.Size(39, 20)**

**Me.timeLabel.TabIndex = 2**

**Me.timeLabel.Text = "time"**

**'**

**'dateLabel**

**'**

**Me.dateLabel.AutoSize = True**

**Me.dateLabel.Location = New System.Drawing.Point(216, 189)**

**Me.dateLabel.Name = "dateLabel"**

**Me.dateLabel.Size = New System.Drawing.Size(45, 20)**

**Me.dateLabel.TabIndex = 1**

**Me.dateLabel.Text = "date "**

**'**

**'CreateAnAppLabel**

**'**

**Me.CreateAnAppLabel.AutoSize = True**

**Me.CreateAnAppLabel.Location = New System.Drawing.Point(76, 59)**

**Me.CreateAnAppLabel.Name = "CreateAnAppLabel"**

**Me.CreateAnAppLabel.Size = New System.Drawing.Size(196, 20)**

**Me.CreateAnAppLabel.TabIndex = 0**

**Me.CreateAnAppLabel.Text = "create an appointment for "**

**'**

**'tabViewAppointment**

**'**

**Me.tabViewAppointment.Controls.Add(Me.Label2)**

**Me.tabViewAppointment.Controls.Add(Me.CreateAppointmentButton)**

**Me.tabViewAppointment.Controls.Add(Me.ListBox1)**

**Me.tabViewAppointment.Location = New System.Drawing.Point(4, 29)**

**Me.tabViewAppointment.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.tabViewAppointment.Name = "tabViewAppointment"**

**Me.tabViewAppointment.Padding = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.tabViewAppointment.Size = New System.Drawing.Size(992, 621)**

**Me.tabViewAppointment.TabIndex = 1**

**Me.tabViewAppointment.Text = "View Appointment"**

**Me.tabViewAppointment.UseVisualStyleBackColor = True**

**'**

**'CreateAppointmentButton**

**'**

**Me.CreateAppointmentButton.Location = New System.Drawing.Point(27, 1054)**

**Me.CreateAppointmentButton.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.CreateAppointmentButton.Name = "CreateAppointmentButton"**

**Me.CreateAppointmentButton.Size = New System.Drawing.Size(2464, 119)**

**Me.CreateAppointmentButton.TabIndex = 1**

**Me.CreateAppointmentButton.Text = "Create Appointment "**

**Me.CreateAppointmentButton.UseVisualStyleBackColor = True**

**'**

**'ListBox1**

**'**

**Me.ListBox1.FormattingEnabled = True**

**Me.ListBox1.ItemHeight = 20**

**Me.ListBox1.Items.AddRange(New Object() {"Existing Appointments:"})**

**Me.ListBox1.Location = New System.Drawing.Point(27, 36)**

**Me.ListBox1.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.ListBox1.Name = "ListBox1"**

**Me.ListBox1.Size = New System.Drawing.Size(965, 444)**

**Me.ListBox1.TabIndex = 0**

**'**

**'tabHome**

**'**

**Me.tabHome.Controls.Add(Me.Button3)**

**Me.tabHome.Controls.Add(Me.Button2)**

**Me.tabHome.Controls.Add(Me.Button1)**

**Me.tabHome.Controls.Add(Me.AboutButton)**

**Me.tabHome.Controls.Add(Me.ViewAppointmentButton)**

**Me.tabHome.Controls.Add(Me.CreateUserSend)**

**Me.tabHome.Controls.Add(Me.ListOfUsers)**

**Me.tabHome.Controls.Add(Me.Label1)**

**Me.tabHome.Location = New System.Drawing.Point(4, 29)**

**Me.tabHome.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.tabHome.Name = "tabHome"**

**Me.tabHome.Padding = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.tabHome.Size = New System.Drawing.Size(992, 621)**

**Me.tabHome.TabIndex = 0**

**Me.tabHome.Text = "Home"**

**Me.tabHome.UseVisualStyleBackColor = True**

**'**

**'Button3**

**'**

**Me.Button3.Location = New System.Drawing.Point(429, 515)**

**Me.Button3.Name = "Button3"**

**Me.Button3.Size = New System.Drawing.Size(139, 55)**

**Me.Button3.TabIndex = 7**

**Me.Button3.Text = "About"**

**Me.Button3.UseVisualStyleBackColor = True**

**'**

**'Button2**

**'**

**Me.Button2.Location = New System.Drawing.Point(787, 515)**

**Me.Button2.Margin = New System.Windows.Forms.Padding(3, 4, 3, 4)**

**Me.Button2.Name = "Button2"**

**Me.Button2.Size = New System.Drawing.Size(172, 55)**

**Me.Button2.TabIndex = 6**

**Me.Button2.Text = "view app"**

**Me.Button2.UseVisualStyleBackColor = True**

**'**

**'Button1**

**'**

**Me.Button1.Location = New System.Drawing.Point(43, 515)**

**Me.Button1.Margin = New System.Windows.Forms.Padding(3, 4, 3, 4)**

**Me.Button1.Name = "Button1"**

**Me.Button1.Size = New System.Drawing.Size(170, 55)**

**Me.Button1.TabIndex = 5**

**Me.Button1.Text = "create app"**

**Me.Button1.UseVisualStyleBackColor = True**

**'**

**'AboutButton**

**'**

**Me.AboutButton.Location = New System.Drawing.Point(983, 1070)**

**Me.AboutButton.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.AboutButton.Name = "AboutButton"**

**Me.AboutButton.Size = New System.Drawing.Size(458, 98)**

**Me.AboutButton.TabIndex = 4**

**Me.AboutButton.Text = "About "**

**Me.AboutButton.UseVisualStyleBackColor = True**

**'**

**'ViewAppointmentButton**

**'**

**Me.ViewAppointmentButton.Location = New System.Drawing.Point(2164, 1070)**

**Me.ViewAppointmentButton.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.ViewAppointmentButton.Name = "ViewAppointmentButton"**

**Me.ViewAppointmentButton.Size = New System.Drawing.Size(314, 98)**

**Me.ViewAppointmentButton.TabIndex = 3**

**Me.ViewAppointmentButton.Text = "View Appointment"**

**Me.ViewAppointmentButton.UseVisualStyleBackColor = True**

**'**

**'CreateUserSend**

**'**

**Me.CreateUserSend.Location = New System.Drawing.Point(54, 1070)**

**Me.CreateUserSend.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.CreateUserSend.Name = "CreateUserSend"**

**Me.CreateUserSend.Size = New System.Drawing.Size(346, 98)**

**Me.CreateUserSend.TabIndex = 2**

**Me.CreateUserSend.Text = "Create User "**

**Me.CreateUserSend.UseVisualStyleBackColor = True**

**'**

**'ListOfUsers**

**'**

**Me.ListOfUsers.FormattingEnabled = True**

**Me.ListOfUsers.ItemHeight = 20**

**Me.ListOfUsers.Location = New System.Drawing.Point(43, 46)**

**Me.ListOfUsers.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.ListOfUsers.Name = "ListOfUsers"**

**Me.ListOfUsers.Size = New System.Drawing.Size(916, 444)**

**Me.ListOfUsers.TabIndex = 1**

**'**

**'Label1**

**'**

**Me.Label1.AutoSize = True**

**Me.Label1.Location = New System.Drawing.Point(39, 22)**

**Me.Label1.Name = "Label1"**

**Me.Label1.Size = New System.Drawing.Size(106, 20)**

**Me.Label1.TabIndex = 0**

**Me.Label1.Text = "Choose User "**

**'**

**'TabControl1**

**'**

**Me.TabControl1.Controls.Add(Me.tabHome)**

**Me.TabControl1.Controls.Add(Me.tabViewAppointment)**

**Me.TabControl1.Controls.Add(Me.tabCreateAppointment)**

**Me.TabControl1.Location = New System.Drawing.Point(25, 28)**

**Me.TabControl1.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.TabControl1.Name = "TabControl1"**

**Me.TabControl1.SelectedIndex = 0**

**Me.TabControl1.Size = New System.Drawing.Size(1000, 654)**

**Me.TabControl1.TabIndex = 0**

**'**

**'Label2**

**'**

**Me.Label2.AutoSize = True**

**Me.Label2.Location = New System.Drawing.Point(431, 539)**

**Me.Label2.Name = "Label2"**

**Me.Label2.Size = New System.Drawing.Size(66, 20)**

**Me.Label2.TabIndex = 2**

**Me.Label2.Text = "Refresh"**

**'**

**'Form1**

**'**

**Me.AutoScaleDimensions = New System.Drawing.SizeF(9.0!, 20.0!)**

**Me.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font**

**Me.ClientSize = New System.Drawing.Size(1051, 712)**

**Me.Controls.Add(Me.TabControl1)**

**Me.Margin = New System.Windows.Forms.Padding(3, 2, 3, 2)**

**Me.Name = "Form1"**

**Me.Text = "MySchedule"**

**Me.tabCreateAppointment.ResumeLayout(False)**

**Me.tabCreateAppointment.PerformLayout()**

**Me.tabViewAppointment.ResumeLayout(False)**

**Me.tabViewAppointment.PerformLayout()**

**Me.tabHome.ResumeLayout(False)**

**Me.tabHome.PerformLayout()**

**Me.TabControl1.ResumeLayout(False)**

**Me.ResumeLayout(False)**

**End Sub**

**Friend WithEvents tabCreateAppointment As TabPage**

**Friend WithEvents TextBox6 As TextBox**

**Friend WithEvents TextBox3 As TextBox**

**Friend WithEvents TextBox2 As TextBox**

**Friend WithEvents ReturnButton As Button**

**Friend WithEvents timeLabel As Label**

**Friend WithEvents dateLabel As Label**

**Friend WithEvents CreateAnAppLabel As Label**

**Friend WithEvents tabViewAppointment As TabPage**

**Friend WithEvents CreateAppointmentButton As Button**

**Friend WithEvents ListBox1 As ListBox**

**Friend WithEvents tabHome As TabPage**

**Friend WithEvents Button2 As Button**

**Friend WithEvents Button1 As Button**

**Friend WithEvents AboutButton As Button**

**Friend WithEvents ViewAppointmentButton As Button**

**Friend WithEvents CreateUserSend As Button**

**Friend WithEvents ListOfUsers As ListBox**

**Friend WithEvents Label1 As Label**

**Friend WithEvents TabControl1 As TabControl**

**Friend WithEvents Button3 As Button**

**Friend WithEvents Label2 As Label**

**End Class**

**Public Class Form1**

**Dim selUser As String = " "**

**Private Sub CreateUserButton\_Click(sender As Object, e As EventArgs)**

**Dim id As Integer**

**Dim name As String**

**Dim address As String**

**Dim x As String = id & "," & name & "," & address**

**Dim wr As New System.IO.StreamWriter("user.txt", True)**

**wr.WriteLine(x)**

**wr.Close()**

**End Sub**

**Private Sub Button1\_Click(sender As Object, e As EventArgs) Handles Button1.Click**

**selUser = ListOfUsers.SelectedItem**

**TabControl1.SelectedTab = tabCreateAppointment**

**Dim s() As String = selUser.Split(",")**

**TextBox6.Text = s(1)**

**End Sub**

**Private Sub ReturnButton\_Click(sender As Object, e As EventArgs) Handles ReturnButton.Click**

**Dim p1 As New Person()**

**p1.name = TextBox6.Text**

**p1.date1 = TextBox2.Text**

**p1.time1 = TextBox3.Text**

**Dim x As String = p1.name & "," & p1.date1 & "," & p1.time1**

**Dim wr As New System.IO.StreamWriter("app.txt", True)**

**wr.WriteLine(x)**

**wr.Close()**

**TabControl1.SelectedTab = tabViewAppointment**

**MessageBox.Show("When you click the button you will" & " be sent to the view appointments tab" & " click below the box to view appointments if any")**

**End Sub**

**Private Sub Form1\_Load(sender As Object, e As EventArgs) Handles MyBase.Load**

**Dim fd() As String = IO.File.ReadAllLines("user.txt")**

**For Each line In fd**

**ListOfUsers.Items.Add(line)**

**Next**

**MessageBox.Show("Click a user to select before pressing a button")**

**End Sub**

**Private Sub tabViewAppointment\_MouseClick(sender As Object, e As MouseEventArgs) Handles tabViewAppointment.MouseClick**

**Dim fd() As String = IO.File.ReadAllLines("app.txt")**

**For Each line In fd**

**ListBox1.Items.Add(line)**

**Next**

**End Sub**

**Private Sub Button2\_Click\_1(sender As Object, e As EventArgs) Handles Button2.Click**

**TabControl1.SelectedTab = tabViewAppointment**

**MessageBox.Show("When you click the button you will" & " be sent to the view appointments tab" & " click below the box to view appointments if any")**

**End Sub**

**Private Sub Button3\_Click(sender As Object, e As EventArgs) Handles Button3.Click**

**MessageBox.Show("Developed in Fall 2019 by" & vbCrLf & vbCrLf & "- Logan Craig" & vbCrLf & "- Gabriel Long" & vbCrLf & "- Luke Fleming" & vbCrLf & "- Daniel Kakwata" & vbCrLf & "- Gartrel Barnes" & vbCrLf & "- Gary Locklear" & vbCrLf & vbCrLf &**

**"This application is released under the terms of GNU General Public license version 3.0. The software comes with absolutely nowarranty.", "About This Application")**

**End Sub**

**End Class**

**Public Class Person**

**Public name As String**

**Public date1 As String**

**Public time1 As String**

**End Class**