College of Engineering, University of Nevada, Reno

Department of Computer Science and Engineering

Computer Science 420

Human-Computer Interaction

FitPets

Project Part 4: Prototype

Authors: Team #1

Nanami Duncan

Pamella Nipay

Jazzel Radaza

Instructor:

Dr. Sergiu Dascalu

13 May 2023

Code Summary

We used the IDE platform Qt Creator to create our phone application prototype. The primary languages used are Python and XML. The key feature of our program is the Widget class, which describes the individual buttons, the pages (interfaces), and transitions for our application.

Main Function:

```
if __name__ == "__main__":
    app = QApplication(sys.argv)
    widget = Widget()
    widget.show()
    sys.exit(app.exec())
```

The main function (located in widget.py) initiates the application and builds all widgets. It builds the application framework, creates the Widget class, and shows the widgets. It keeps running the program until the user makes the exit command. The following libraries are included: PySide6.QtWidgets, sys, and Ui_Widget.

Primary Class - Widget

```
class Widget(QWidget):
    def __init__(self, parent=None):
        super().__init__(parent)
        self.ui = Ui_Widget()
        self.ui.setupUi(self)
```

This class defines what a widget is. In our program, it defines all buttons, page transitions, and text fields in our app prototype. The image above is the class declaration and widget initialization parameters.

Widget Type - pushButton#

```
self.start= self.findChild(QPushButton, "pushButton")
self.next= self.findChild(QPushButton, "pushButton_2")
self.plus= self.findChild(QPushButton, "plus")
self.stacked= self.findChild(QStackedWidget, "stackedWidget")
                            ⋮
self.login= self.findChild(QPushButton, "pushButton_6")
self.log= self.findChild(QPushButton, "pushButton_7")
self.day= self.findChild(QPushButton, "pushButton_8")
self.add= self.findChild(QPushButton, "pushButton_9")
self.back= self.findChild(QPushButton, "pushButton_10")
self.register= self.findChild(QPushButton, "pushButton_11")
```

These widgets are the buttons found throughout the app. Pushing the button initiates the lambda + self functions listed at the end of the Widget class description. The name of the button (self.name. …) represents the action the button performs when tapped (e.g., “next” sends the current screen to the next screen).

Widget Type - page#

```
                    ⋮
self.page1= self.findChild(QWidget, "page")
self.page2= self.findChild(QWidget, "page_2")
self.page3= self.findChild(QWidget, "page_3")
self.page4= self.findChild(QWidget, "page_4")
self.page5= self.findChild(QWidget, "page_5")
self.page6= self.findChild(QWidget, "page_6")
self.page7= self.findChild(QWidget, "page_7")
self.page8= self.findChild(QWidget, "page_8")
self.page8= self.findChild(QWidget, "page_9")
                    ⋮
```

These widgets are the pages of our prototype app, such as the home page, exercise log, costume shop, etc.

Widget Type - lineEdit#

```
self.lineEdit= self.findChild(QLineEdit, "lineEdit")
self.lineEdit.setPlaceholderText("Email or Phone Number")
self.lineEdit.setEchoMode(QLineEdit.EchoMode.Normal)

self.lineEdit2= self.findChild(QLineEdit, "lineEdit_2")
self.lineEdit2.setPlaceholderText("Username")
self.lineEdit2.setEchoMode(QLineEdit.EchoMode.Normal)

self.lineEdit3= self.findChild(QLineEdit, "lineEdit_3")
self.lineEdit3.setPlaceholderText("Full Name")
self.lineEdit3.setEchoMode(QLineEdit.EchoMode.Normal)

self.lineEdit4= self.findChild(QLineEdit, "lineEdit_4")
self.lineEdit4.setPlaceholderText("Password")
self.lineEdit4.setEchoMode(QLineEdit.EchoMode.Normal)

self.lineEdit5= self.findChild(QLineEdit, "lineEdit_5")
self.lineEdit5.setPlaceholderText("Confirm Password")
self.lineEdit5.setEchoMode(QLineEdit.EchoMode.Normal)
```

These widgets are the text entry boxes found on some pages in the app. The specific text boxes in our prototype pertain to the first login page (email, username, actual name, password, and confirm password).

Lambda + Self Functions

```
self.start.clicked.connect(lambda: self.stacked.setCurrentWidget(self.page2))
self.next.clicked.connect(lambda: self.stacked.setCurrentWidget(self.page3))
self.plus.clicked.connect(lambda: self.stacked.setCurrentWidget(self.page4))
self.login.clicked.connect(lambda: self.stacked.setCurrentWidget(self.page5))
self.log.clicked.connect(lambda: self.stacked.setCurrentWidget(self.page6))
self.day.clicked.connect(lambda: self.stacked.setCurrentWidget(self.page7))
self.add.clicked.connect(lambda: self.stacked.setCurrentWidget(self.page8))
self.back.clicked.connect(lambda: self.stacked.setCurrentWidget(self.page6))
self.register.clicked.connect(lambda: self.stacked.setCurrentWidget(self.page9))
```

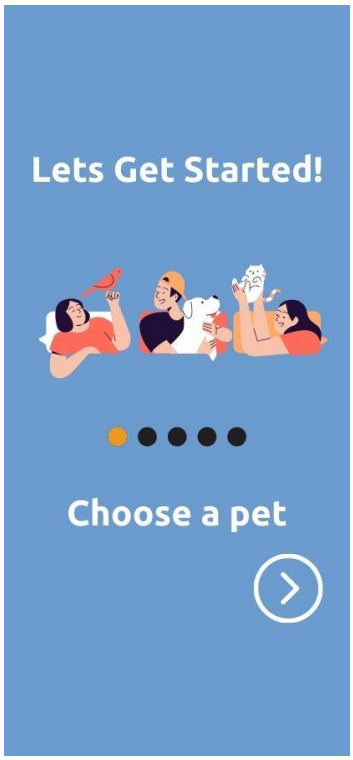
These functions connect the various pages of our prototype app pages together. For example, selecting the “Start” button changes the current screen to that of the page called “page2.”

Second Class - UI_Widget

This class (defined in the Python file named “ui_form.py) further defines the widgets. This class defines the widgets' appearance (color, border, padding, font size, etc.) on a selected screen. The class also defines the actions executed when a widget is interacted with. To demonstrate, I’ve selected one of the pages (code below):

```
self.stackedWidget.addWidget(self.page)
self.page_2 = QWidget()
self.page_2.setObjectName(u"page_2")
self.label_2 = QLabel(self.page_2)
self.label_2.setObjectName(u"label_2")
self.label_2.setGeometry(QRect(-20, -20, 321, 581))
self.label_2.setPixmap(QPixmap(u"Static images/Static Image version 3.2.jpg"))
self.label_2.setScaledContents(True)
self.pushButton_2 = QPushButton(self.page_2)
self.pushButton_2.setObjectName(u"pushButton_2")
self.pushButton_2.setGeometry(QRect(230, 410, 31, 41))
self.pushButton_2.setStyleSheet(u"QPushButton {\n"
```

This section defines the appearance and sets the widgets on the starting page when the user first downloads the app:



Team Contributions

Team Member	Contribution	Hours
Nanami Duncan	Code summary, Demo video (audio, script)	3
Pamella Nipay	Prototype code, Demo video (video)	8.5
Jazzel Radaza	Prototype code, video script, Interface designs, Demo video (script)	6.5