

TRƯỜNG ĐẠI HỌC MỞ THÀNH PHỐ HỒ CHÍ MINH
KHOA CÔNG NGHỆ THÔNG TIN



NGUYỄN THỊ BÍCH TUYÊN

TÊN BÁO CÁO THỰC TẬP TỐT NGHIỆP

BÁO CÁO THỰC TẬP TỐT NGHIỆP

Ngành: Công Nghệ Thông Tin

Hướng chuyên ngành: Khoa học máy tính

Lớp: DH15TH01

NGƯỜI HƯỚNG DẪN KHOA HỌC
Ths. Dương Hữu Thành

Thành phố Hồ Chí Minh – Năm 2018

Lời cảm ơn

Sau khi kết thúc kì thực tập vô cùng bổ ích, em đã được trải nghiệm và học hỏi rất nhiều. Để em có được những điều đáng quý trên, chắc chắn không thể nào vắng bóng thầy cô, quý công ty và đặc biệt là người hướng dẫn em trong suốt quá trình thực tập.

Đầu tiên em xin gửi lời cảm ơn chân thành đến nhà trường – Đại học Mở Thành phố Hồ Chí Minh đã tạo điều kiện cho em có cơ hội và kiến thức nền tảng để đi thực tập.

Em cũng xin gửi lời cảm ơn đến thầy Dương Hữu Thành, người đã đồng hành cùng em trong suốt những năm học qua. Thầy đã chỉ bảo, quan tâm cho em những lời khuyên để em hoàn thành bài báo cáo.

Cuối cùng em cảm thấy rất vui và hạnh phúc khi được vào thực tập ở công ty Robert Bosch Engineering Việt Nam. Mọi người luôn sẵn lòng hỗ trợ và giải đáp những thắc mắc của em.

Em thật sự cảm ơn mọi người đã luôn sát cánh bên em trong suốt quá trình em thực tập, để em có được nhiều kinh nghiệm quý giá làm hành trang để hướng đến tương lai.

[illegible]

3

Mục lục hình ảnh

Hình 1: Thông tin về Robert Bosch Việt Nam

Hình 2: Giải pháp Mobility

Hình 3: Phụ tùng và phụ kiện ô tô

Hình 4: Thiết bị gia dụng

Hình 5: Công nghiệp & Thương mại

Hình 6: Kiến trúc lớp Qwidget

Hình 7: Chức năng của các lớp thường được sử dụng trong Qt

Hình 8: Giao diện Qt Designer

Hình 9: Tạo 1 form bằng Qt Designer

Hình 10: Tạo 1 form bằng dòng lệnh

Mục lục

Chương 1: TỔNG QUAN	6
1. Lịch làm việc tại nơi thực tập.....	6
2. Giới thiệu về tổ chức của nơi thực tập	6
2.1. Giới thiệu về Robert Bosch VietNam	6
2.2. Các lĩnh vực hoạt động của Robert Bosch VietNam:	6
2.3. Thông tin liên hệ:.....	9
3. Nội dung nhiệm vụ chính được giao	9
Chương II: NỘI DUNG CÁC CÔNG VIỆC VÀ KẾT QUẢ ĐẠT ĐƯỢC	10
1. Đặt vấn đề.....	10
2. Cách tiếp cận	10
3. Giới thiệu đề tài	10
4. Nội dung đề tài	10
4.1. Pyside	11
4.1.1. Giới thiệu về Pyside	11
Cài đặt Pyside.....	11
Lưu ý	11
4.1.2. Kiến trúc các lớp trong Qt:.....	11
4.1.3. Signal và slot	14
Event	14
Signal	14
Slot	14
4.1.4. Các widget cơ bản	14
4.1.5. Cách tạo một GUI Pyside	14
Thông qua giao diện:.....	14
Thông qua dòng lệnh:	16
4.2. Làm việc với file/folder trong python.....	16
4.2.1. Các định nghĩa trong xử lý file	16
4.2.2. Đọc file.....	17
4.2.3. Ghi/Sửa file	17
4.2.4. Xóa file.....	17
4.2.5. Sao chép file.....	18
4.2.6. Đọc folder.....	18
4.2.7. Xóa folder.....	18
4.2.8. Sao chép folder.....	18
CHƯƠNG III: KẾT QUẢ ĐẠT ĐƯỢC QUA ĐỢT THỰC TẬP	19
1. Những kiến thức lý thuyết đã được củng cố	19
2. Những kỹ năng thực hành đã được học thêm	19
3. Những kinh nghiệm thực tiễn đã tích lũy được.....	19

Chương 1: TỔNG QUAN

1. Lịch làm việc tại nơi thực tập

Buổi	Thứ 2	Thứ 3	Thứ 4	Thứ 5	Thứ 6
Sáng	9h – 12h	9h – 12h	9h – 12h	9h – 12h	9h – 12h
Chiều	13h – 18h	13h – 18h	13h – 18h	13h – 18h	13h – 18h

Tháng đầu tiên, training vào sáng thứ 4 hằng tuần.

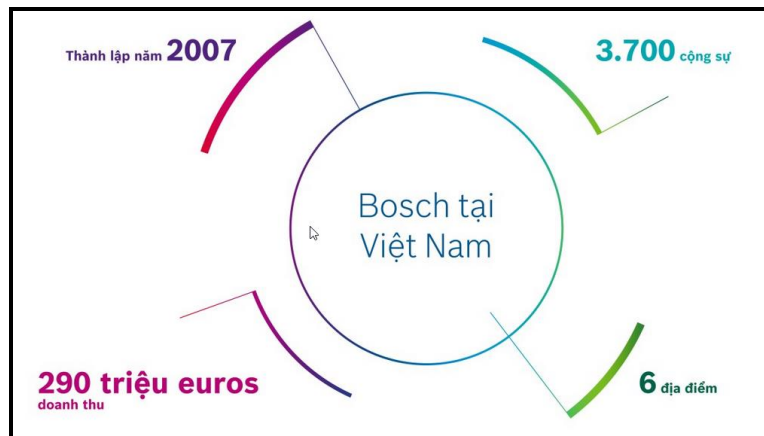
2. Giới thiệu về tổ chức của nơi thực tập

2.1. Giới thiệu về Robert Bosch Việt Nam

Bosch bắt đầu hoạt động tại Việt Nam năm 1994. Công ty TNHH Bosch Việt Nam có trụ sở chính tại TP. Hồ Chí Minh, hai văn phòng chi nhánh ở Hà Nội và Đà Nẵng cùng nhà máy Gasoline Systems sản xuất dây đai truyền lực biến đổi liên tục (CVT) tại tỉnh Đồng Nai.

Ngoài ra, Bosch cũng vận hành một trung tâm Nghiên cứu và Phát triển Công nghệ và các giải pháp doanh nghiệp (Công ty TNHH Robert Bosch Engineering & Business Solutions Việt Nam) và Trung tâm Nghiên cứu & Phát triển Công nghệ Ô tô tại TP. Hồ Chí Minh.

Tính đến 31/12/2017, Bosch đã tuyển dụng hơn 3.700 cộng sự tại Việt Nam.



Hình 1: Thông tin về Robert Bosch Việt Nam

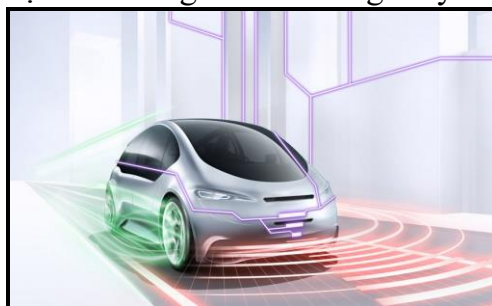
2.2. Các lĩnh vực hoạt động của Robert Bosch Việt Nam:

Tại Việt Nam, Bosch có các ngành hàng kinh doanh: Phụ tùng và Thiết bị Ô tô; Điện tử Ô tô và Xe máy; Kỹ thuật Truyền động và Điều khiển; Kỹ thuật Đóng gói; Dụng cụ Điện cầm tay; và Công nghệ Tòa nhà.

Giải pháp Mobility

Cổng thông tin giải pháp mobility của Bosch cung cấp các nội dung chính về giải pháp mobility kết nối, giải pháp mobility tự động, động cơ và giải pháp mobility

điện. Các sản phẩm và dịch vụ của chúng tôi không chỉ ứng dụng cho ô tô mà còn cho xe tải, xe hai bánh, vận tải đường sắt và đường thủy.

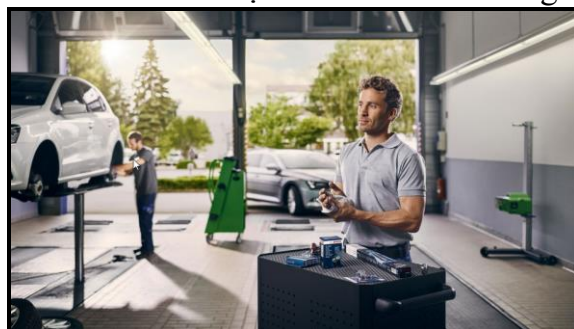


Hình 2: Giải pháp Mobility

Phụ tùng và phụ kiện ô tô

Bosch giúp bạn luôn chuyển động.

Phụ tùng ô tô Bosch nổi tiếng về chất lượng, độ tin cậy và công nghệ tiên tiến. Hệ thống chiếu sáng và phanh hiện đại, ắc quy có tuổi thọ dài, cần gạt nước, bộ khởi động và bộ lọc không khí — Bosch cung cấp phụ tùng chất lượng cao giúp lái xe an toàn hơn, thoải mái hơn và thân thiện hơn với môi trường.



Hình 3: Phụ tùng và phụ kiện ô tô

Thiết bị gia dụng

Dụng cụ làm vườn

Bạn đang tìm kiếm giải pháp cho việc vệ sinh xe hay hồ bơi? Bạn muốn dọn dẹp sân vườn nhanh chóng hay rửa sạch những vết bánh xe đầy bùn đất? Bất kể bạn cần gì, Bosch đều mang đến nhiều lựa chọn đa dạng với những dòng máy phun xịt rửa áp lực cao — thích hợp cho mọi mục đích xịt rửa từ đơn giản đến phức tạp.

Dụng cụ điện cầm tay cho người mê DIY

Có trong tay sản phẩm phù hợp, việc tự chế tác sẽ thú vị gấp đôi. Với Bosch, bạn không chỉ có nhiều lựa chọn hơn, mà còn có thể làm việc chính xác, linh động, an toàn và hiệu quả hơn. Dụng cụ của chúng tôi có chất lượng cao, khả năng xử lý ưu việt và công nghệ tiên tiến. Dù là công việc chế tác nào, bạn đều có thể làm được với dụng cụ điện cầm tay của Bosch.



Hình 4: Thiết bị gia dụng

Công nghiệp & Thương mại

Kỹ thuật Truyền động & Điều Khiển

Bosch là đối tác năng động trên toàn cầu, với danh mục sản phẩm độc đáo dành cho các ứng dụng di động, tự động hóa nhà máy và nhà máy chế biến, tiêu biểu cho công nghệ tiên tiến và chuyên môn vượt trội trong ngành. Các cộng sự giàu kinh nghiệm của chúng tôi làm việc không ngừng để phát triển các giải pháp an toàn, linh hoạt và tiết kiệm tài nguyên trên toàn cầu. Các nhà sản xuất máy móc và người dùng cuối đều được hưởng lợi từ những cải tiến đa chức năng của chúng tôi. Những cải tiến này cũng được thiết kế để sử dụng trong môi trường sản xuất kết nối.

Kỹ thuật đóng gói

Bosch là một trong những nhà cung cấp hàng đầu cho ngành công nghiệp sản xuất và đóng gói. Tại hơn 30 địa điểm ở hơn 15 quốc gia trên toàn cầu, lực lượng lao động có trình độ cao đã phát triển và đưa ra thị trường các giải pháp hoàn chỉnh cho ngành dược, thực phẩm và bánh kẹo. Từ các giải pháp tiêu chuẩn tới các dây chuyền hoàn chỉnh và các hệ thống tích hợp, chúng tôi phát triển thiết bị với phương châm “Sáng tạo vì cuộc sống”.

Dụng cụ điện cầm tay chuyên nghiệp

Bosch phát triển các sản phẩm dụng cụ điện cầm tay tiên tiến, chất lượng và hiệu quả cao. Công cụ của chúng tôi đáng tin cậy, mạnh mẽ, đáp ứng được những yêu cầu cao nhất và giúp bạn làm việc dễ dàng hơn. Các công cụ không dây của Bosch nổi bật với độ bền ưu việt, khả năng hoạt động tốt với cường độ cao và thời gian sử dụng dài.

Các giải pháp an ninh

Bảo vệ sự sống, tòa nhà, tài sản là mục tiêu của chúng tôi. Danh mục sản phẩm của chúng tôi gồm có hệ thống giám sát bằng video, phát hiện xâm nhập, phát hiện hỏa hoạn, các hệ thống sơ tán bằng giọng nói cũng như các hệ thống quản lý và kiểm soát ra vào. Bên cạnh đó, chúng tôi còn có hệ thống âm thanh biểu diễn và hội nghị chuyên nghiệp.

Các giải pháp phần mềm

Internet vạn vật đang thay đổi thế giới và sẽ mang lại nhiều cơ hội cho các lĩnh vực như sản xuất, giao thông, quản lý năng lượng và nhiều ngành khác. Các phần mềm và dịch vụ Bosch cung cấp giúp tạo nền tảng vững chắc cho

việc cài đặt và vận hành các giải pháp chuyên biệt. Khám phá cách các giải pháp phần mềm của Bosch hỗ trợ doanh nghiệp cải thiện việc điều hành kinh doanh và mở ra nhiều kênh doanh thu và sản phẩm mới.



Hình 5: Công nghiệp & Thương mại

2.3. Thông tin liên hệ:

Công ty TNHH Robert Bosch Engineering and Business Solutions Việt Nam
Lầu 11, Tòa nhà E-town 2, 364 Cộng Hòa, Quận Tân Bình, Thành phố Hồ Chí Minh, Việt Nam
Điện thoại (84 28) 3812 8000
Fax (84 28) 3812 8001
Email www.bosch-india-software.com

3. Nội dung nhiệm vụ chính được giao

Tuần 1:

Đọc và tìm hiểu tài liệu liên quan đến thao tác trên thư mục bằng Python.
Tiếp cận với project thực tế đã đưa vào sử dụng.

Tuần 2:

Tìm hiểu thư viện mà project sử dụng: Pyside, os, sys,...

Tuần 3:

Tìm hiểu công nghệ được sử dụng trong project như: ldap, MS outlook,...

Tuần 4:

Tiến hành thu thập thông tin yêu cầu để xây dựng chương trình sao chép file bằng Python.

Tuần 5:

Phát thảo giao diện và chức năng chương trình.
Tiến hành “flash” chương trình trên target.

Tuần 6:

Tìm hiểu các thư viện copy file như: shutil.
Tiến hành lập trình chương trình “copy tool”.

Tuần 7:

Thích ứng với việc thay đổi yêu cầu từ khách hàng.
Tìm hiểu về đa luồng trong một chương trình.
Bổ sung, chỉnh sửa những chức năng để hoàn thiện hơn.

Tuần 8:

Demo và tổng kết quá trình hoàn thiện chương trình.

Chương II: NỘI DUNG CÁC CÔNG VIỆC VÀ KẾT QUẢ ĐẠT ĐƯỢC

1. Đặt vấn đề

Kinh tế, xã hội ngày càng phát triển, đòi hỏi nhu cầu ứng dụng của công nghệ thông tin vào các lĩnh vực đời sống ngày càng cao và không ngừng thay đổi. Khi hệ thống ngày càng phát triển thì yêu cầu tự động hóa càng được chú trọng để có thể đáp ứng được nhu cầu càng cao của hệ thống. Lập trình viên phải luôn tự đặt câu hỏi làm sao để có thể tăng tốc độ và hỗ trợ bản thân thực hiện những công việc đơn giản mà lại tốn rất nhiều thời gian, đó là việc “copy – paste”. Việc sao chép dữ liệu hầu như diễn hằng ngày và nó cứ lặp đi lặp lại, khiến chúng ta mất thời gian chờ đợi vì đa số các project thường chứa ít nhất vài chục gigabyte. Vì vậy việc tự động hóa công việc trên là hoàn toàn cần thiết. Nó giúp tiết kiệm thời gian, hỗ trợ người lập trình tìm kiếm và nâng cao năng suất công việc cho công ty.

2. Cách tiếp cận

Dựa vào project “flash station” đã được xây dựng từ trước, tiến hành đọc tài liệu có liên quan và tìm hiểu cách thức hoạt động của project trên. Ta có thể tự xây dựng một chương trình “Copy tool” với quy mô nhỏ hơn, phù hợp với môi trường và nhu cầu làm việc hiện tại.

3. Giới thiệu đề tài

“Copy tool” là một chương trình do em xây dựng dựa trên yêu cầu thực tế tại công ty. Nó là mô phỏng nhỏ cho một chức năng trong project “flash station”. Chức năng chính là hỗ trợ việc copy file/folder, giúp tự động hóa việc sao chép dữ liệu. Chương trình sử dụng ngôn ngữ Python, tìm và xác định địa chỉ file thông qua thư viện os, sao chép file/folder thông qua thư viện shutil, tạo giao diện nhờ thư viện Pyside. Ngoài ra để tạo progress bar em áp dụng multithread trong Python. Chương trình sử dụng design pattern là MVC model kết hợp với thư viện configparser hỗ trợ quản lý, chỉnh sửa khi có thay đổi. Dữ liệu được lưu trữ trong thư viện MySQL và liên kết với chương trình thông qua framework python-mysql.

4. Nội dung đề tài

Trong suốt 8 tuần thực tập, em đã học được rất nhiều kinh nghiệm và kiến thức ở công ty. Em sẽ tóm tắt 2 nội dung chính em học được là thư viện Pyside và làm việc trên file/folder trong Python:

Pyside:

Giới thiệu về Pyside

Kiến thức các lớp trong Qt

Signal và slot

Các widget cơ bản

Cách tạo 1 GUI Pyside

Làm việc trên file/folder trong Python:

Đọc, ghi, xóa, copy file.
Đọc, ghi, xóa, copy folder.

4.1. Pyside

4.1.1. Giới thiệu về Pyside

PySide là thư viện Python để tạo giao diện người dùng đồ họa đa nền tảng. Nó là một ràng buộc Python với khung công tác Qt. Thư viện Qt là một trong những thư viện GUI mạnh mẽ nhất.

PySide được thực hiện như một tập hợp các mô-đun Python. Hiện tại nó có 15 mô-đun. Các mô-đun này cung cấp các công cụ mạnh mẽ để làm việc với GUI, đa phương tiện, tài liệu XML, mạng hoặc cơ sở dữ liệu.

Mô-đun QtCore chứa chức năng cốt lõi trong GUI. Mô-đun này được sử dụng để làm việc với thời gian, tệp và thư mục, các loại dữ liệu, luồng, URL, loại mime, chủ đề hoặc quy trình khác nhau. Module QtGui chứa các thành phần đồ họa và các lớp liên quan. Chúng bao gồm các nút ví dụ, cửa sổ, thanh trạng thái, thanh công cụ, thanh trượt, bitmap, màu sắc, phong chữ,...

PySide đã được phát hành sau Nokia, chủ sở hữu bộ công cụ Qt, đã không đạt được thỏa thuận với Riverbank Computing để đưa LGPL như một giấy phép thay thế. PySide có khả năng tương thích API cao với PyQt4, vì vậy việc di chuyển sang PySide không khó.

Cài đặt Pyside

Window: pip install PySide

Linux(Ubuntu): sudo apt-get install python-pyside
sudo apt-get install python-qt4

Lưu ý

PySide và PyQt có thể tương thích API với nhau (có thể dùng chung các class), nên những project sử dụng PyQt có thể chuyển sang PySide và ngược lại.

PySide chưa hỗ trợ python 3 cho đến phiên bản 1.0.8, hỗ trợ tốt hơn cho python2. PyQt hỗ trợ tốt cho python 3.

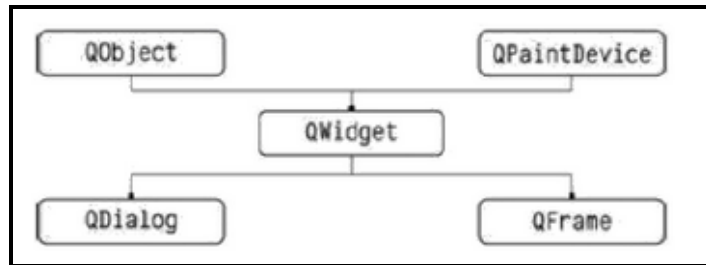
Nên tùy vào nhu cầu sử dụng ta sẽ chọn framework cho phù hợp.

4.1.2. Kiến trúc các lớp trong Qt:

Lớp QObject là lớp cơ sở của tất cả các đối tượng Qt. Ngoài ra, lớp QPaintDevice là lớp cơ sở cho tất cả các đối tượng có thể được vẽ.

Class QApplication quản lý các thiết lập chính và luồng điều khiển của một ứng dụng GUI. Nó chứa vòng lặp sự kiện chính bên trong đó các sự kiện được tạo bởi các cửa sổ và các nguồn khác được xử lý và gửi đi. Nó cũng xử lý các thiết lập trên toàn hệ thống và ứng dụng.

Class QWidget, bắt nguồn từ các lớp QObject và QPaintDevice là lớp cơ sở cho tất cả các đối tượng giao diện người dùng. Các lớp QDialog và QFrame cũng được bắt nguồn từ lớp QWidget.



Hình 6:Kiến trúc lớp Qwidget

Sr.No.	Modules & Description
1	QtCore Core non-GUI classes used by other modules
2	QtGui Graphical user interface components
3	QtMultimedia Classes for low-level multimedia programming
4	QtNetwork Classes for network programming
5	QtOpenGL OpenGL support classes
6	QtScript Classes for evaluating Qt Scripts
7	QtSql Classes for database integration using SQL
8	QtSvg Classes for displaying the contents of SVG files
9	QtWebKit Classes for rendering and editing HTML
10	QtXml Classes for handling XML
11	QtAssistant Support for online help
12	QtDesigner Classes for extending Qt Designer

Hình 7: Chức năng của các lớp thường được sử dụng trong Qt

4.1.3. Signal và slot

Event

Các hàm hoặc phương thức được thực hiện để phản hồi lại hành động của người dùng như nhấp vào nút, chọn một mục từ bộ sưu tập hoặc nhấp chuột,..được gọi là sự kiện.

Ví dụ: `b1.clicked.connect(b1_clicked())`

Signal

Mỗi widget PySide, được bắt nguồn từ lớp `QObject`, được thiết kế để phát ra 'signal' đáp ứng một hoặc nhiều sự kiện.

Slot

Tín hiệu của riêng của sự kiện không thực hiện bất kỳ hành động nào. Thay vào đó, nó được 'kết nối' với một **slot**. **Slot** có thể là bất kỳ chức năng Python có thể gọi nào.

Ví dụ: `QtCore.QObject.connect(widget, QtCore.SIGNAL('signalname'), slot_function)`

4.1.4. Các widget cơ bản

QLabel: Hiển thị văn bản hoặc hình ảnh không thể chỉnh sửa hoặc phim GIF động.

QPushButton: Hiển thị nút mà khi được bấm có thể được lập trình để gọi một hàm nhất định.

QRadioButton: Hiển thị một nút có thể lựa chọn với một nhãn văn bản.

QCheckBox: Hiển thị một nút có thể lựa chọn với một nhãn văn bản, có thể lựa chọn nhiều option cùng 1 lúc.

QComboBox: Hiển thị một danh sách thả xuống các mục để chọn.

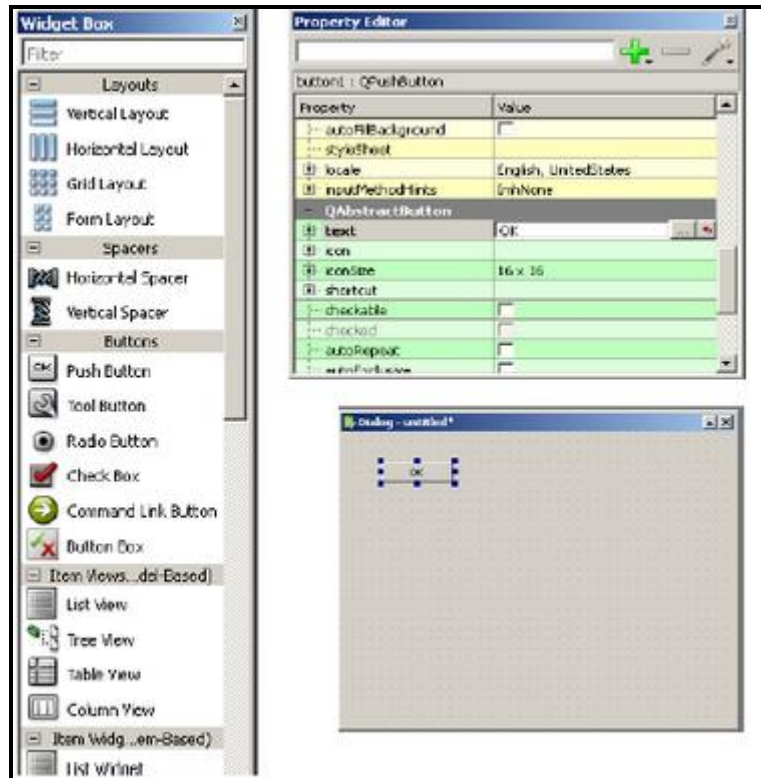
4.1.5. Cách tạo một GUI PySide

Thông qua giao diện:

Bộ cài đặt PyQt đi kèm với một công cụ xây dựng GUI được gọi là Qt Designer. Sử dụng giao diện kéo và thả đơn giản của nó, một giao diện GUI có thể được xây dựng nhanh chóng mà không cần phải viết mã. Tuy nhiên, không phải là một IDE như Visual Studio. Do đó, Qt Designer không có cơ sở để gỡ lỗi và xây dựng ứng dụng.

Tạo một giao diện GUI bằng cách sử dụng Qt Designer bắt đầu bằng cách chọn một cửa sổ cấp cao nhất cho ứng dụng.

Sau đó, bạn có thể kéo và thả các tiện ích bắt buộc từ hộp tiện ích ở ngăn bên trái. Bạn cũng có thể gán giá trị cho các thuộc tính của widget được đặt trên biểu mẫu.



Hình 8: Giao diện Qt Designer

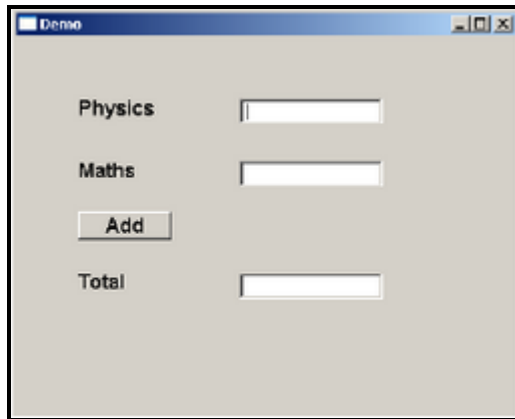
Biểu mẫu được thiết kế được lưu dưới dạng demo.ui. Tập ui này chứa biểu diễn XML của các widget và các thuộc tính của chúng trong thiết kế. Thiết kế này được dịch sang Python tương đương bằng cách sử dụng tiện ích dòng lệnh pyuic4. Tiện ích này là một trình bao bọc cho mô-đun uic. Việc sử dụng pyuic4 như sau:

pyuic4 -x demo.ui -o demo.py

Trong lệnh trên, công tắc -x thêm một số lượng nhỏ mã bổ sung vào XML được tạo ra để nó trở thành một ứng dụng độc lập tự thực thi.

```
if __name__ == "__main__":
    import sys
    app = QtGui.QApplication(sys.argv)
    Dialog = QtGui.QDialog()
    ui = Ui_Dialog()
    ui.setupUi(Dialog)
    Dialog.show()
    sys.exit(app.exec_())
```

Ta sẽ nhận được kết quả:



Hình 9: Tạo 1 form bằng Qt Designer

Thông qua dòng lệnh:

Tạo một ứng dụng GUI đơn giản bằng PyQt bao gồm các bước sau:

Nhập mô-đun QtGui.

Tạo một đối tượng ứng dụng.

Một đối tượng QWidget tạo cửa sổ cấp cao nhất. Thêm đối tượng QLabel vào.

Đặt chú thích của nhãn là "hello world".

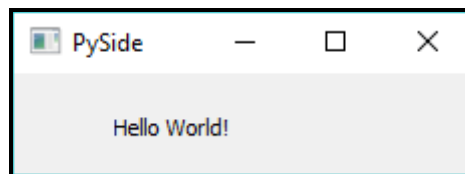
Xác định kích thước và vị trí của cửa sổ bằng phương thức setGeometry ().

Nhập mainloop của ứng dụng bằng phương thức app.exec_ ().

```
import sys
from PySide import QtGui

def window():
    app = QtGui.QApplication(sys.argv)
    w = QtGui.QWidget()
    b = QtGui.QLabel(w)
    b.setText("Hello World!")
    w.setGeometry(100,100,200,50)
    b.move(50,20)
    w.setWindowTitle('PySide')
    w.show()
    sys.exit(app.exec_())

if __name__ == '__main__':
    window()
```



Hình 10: Tạo 1 form bằng dòng lệnh

4.2. Làm việc với file/folder trong python

4.2.1. Các định nghĩa trong xử lý file

Chức năng quan trọng để làm việc với các tệp trong Python là hàm `open()`.

Hàm `open()` nhận hai tham số: tên tệp và phương thức.

Có bốn phương thức (chế độ) khác nhau để mở tệp:

"r" - Đọc - Giá trị mặc định. Mở tệp để đọc, lỗi nếu tệp không tồn tại.

"a" - Append - Mở tệp để thêm, tạo tệp nếu nó không tồn tại.

"w" - Viết - Mở một tệp để ghi, tạo tệp nếu nó không tồn tại.

"x" - Tạo - Tạo tệp được chỉ định, trả về lỗi nếu tệp tồn tại.

Ngoài ra, bạn có thể chỉ định xem tệp có nên được xử lý dưới dạng chế độ văn bản hoặc nhị phân hay không:

"t" - Văn bản - Giá trị mặc định. Chế độ văn bản.

"b" - Nhị phân - Chế độ nhị phân (ví dụ: hình ảnh, file .exe)

4.2.2. Đọc file

Để mở tệp, hãy sử dụng hàm `open()` tích hợp sẵn.

Hàm `open()` trả về một đối tượng tệp, có một phương thức `read()` để đọc nội dung của tệp:

```
f = open("demofile.txt", "r")
print(f.read())
```

Chỉ đọc các phần của tệp

Theo mặc định, phương thức `read()` trả về toàn bộ văn bản, nhưng ta cũng có thể chỉ định số ký tự ta muốn trả về:

```
f = open("demofile.txt", "r")
print(f.read(5))
```

Đọc theo dòng

Ta có thể trả về một dòng bằng cách sử dụng phương thức `readline()`:

```
f = open("demofile.txt", "r")
print(f.readline())
```

4.2.3. Ghi/Sửa file

Để ghi vào một tệp hiện có, bạn phải thêm một tham số vào hàm `open()`:

"a" - Append - sẽ nối vào cuối tệp.

"w" - Write - sẽ ghi đè bất kỳ nội dung hiện có nào.

```
f = open("demofile.txt", "a")
f.write("Now the file has one more line!")
f = open("demofile.txt", "w")
f.write("Woops! I have deleted the content!")
```

Để tạo một tệp mới bằng Python, hãy sử dụng phương thức `open()`, với một trong các tham số sau:

"x" - Create - sẽ tạo một tệp, trả về lỗi nếu tệp tồn tại.

"a" - Append - sẽ tạo một tệp nếu tệp được chỉ định không tồn tại.

"w" - Write - sẽ tạo một tệp nếu tệp được chỉ định không tồn tại.

```
f = open("myfile.txt", "x")
```

4.2.4. Xóa file

Để xóa một tệp, ta phải nhập mô-đun OS và chạy hàm `os.remove()`:

```
import os
os.remove("demofile.txt")
```

4.2.5. Sao chép file

Ta sử dụng thư viện Shutil, để sao chép file ta sử dụng hàm `shutil.copy()` (“tên file”, “địa chỉ copy đến”)

```
import shutil
def main():
    try:
        shutil.copy("abc.txt", '/media')
    except IOError as e:
        print "I/O error({0})".format( e.strerror)
```

4.2.6. Đọc folder

Ta có thể đọc tên của 1 folder bằng cách sử dụng thư viện “os”:

os.listdir() (“địa chỉ”): trả về list tên của folder.

```
import os
for dic_name in os.listdir(root):
    print dic_name
```

Ngoài ra ta có thể lấy được path của folder, tên folder và tên file có trong 1 địa chỉ bằng các sử dụng phương thức **walk** trong thư viện os.

```
from os import walk
f = []
for (dirpath, dirnames, filenames) in walk(mypath):
    f.extend(filenames)
```

4.2.7. Xóa folder

Để xóa toàn bộ thư mục, sử dụng phương thức `os.rmdir()`:

```
import os
os.rmdir("myfolder")
```

Sử dụng `shutil`:

```
shutil.rmtree("địa chỉ")
```

4.2.8. Sao chép folder

Dùng thư viện os:

```
with open(src, 'rb') as fsrc:
    with open(dest, 'wb') as fdst:
        while 1:
            buf = fsrc.read(1024)
            if not buf:
                break
            fdst.write(buf)
```

Dùng thư viện `shutil`:

```
shutil.copytree(src, dest)
```

CHƯƠNG III: KẾT QUẢ ĐẠT ĐƯỢC QUA ĐỢT THỰC TẬP

1. Những kiến thức lý thuyết đã được củng cố

Củng cố kiến thức ngôn ngữ Python

Tìm hiểu thêm được nhiều thư viện hỗ trợ Python

Hiểu rõ hơn về mô hình MVC

Cách sử dụng git và các mô hình quản lý source code.

Làm quen được với những phân cứng “target” trong mô tô.

2. Những kỹ năng thực hành đã được học thêm

Kỹ năng lập trình python.

Khả năng tư duy và giải quyết vấn đề.

3. Những kinh nghiệm thực tiễn đã tích lũy được

Tìm hiểu và tiếp cận được quy trình làm việc chuyên nghiệp.

Môi trường và thái độ khi làm việc.

Khả năng giải quyết và hỏi đáp khi có vấn đề xảy ra.

Tầm quan trọng của tiếng anh và tính cẩn thận.

Ghi nhớ báo cáo mỗi ngày trước khi xong việc.

Tài liệu tham khảo:

https://www.tutorialspoint.com/pyqt/pyqt_hello_world.htm

<https://docs.python.org/3/library/os.html>

<https://docs.python.org/2/library/shutil.html>

<https://www.w3schools.com/python>