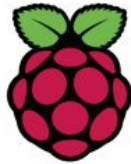


Open source

WEB KIOSK

beta



POWERED BY
Raspberry Pi

A red wavy line with a small Raspberry Pi logo at the end, located at the bottom of the rectangular box.

Topics

1. What was Our Project?
2. Idea?
3. Motive?
4. End Product?
5. How?
6. Code?
7. Manual
8. Challenges
9. Improvements



Open Source Web Kiosk?

- Information Displayer (websites, photos, slideshows)
- Personalize
- Locked in Full Screen
- Automatic on Start-up

Found in:

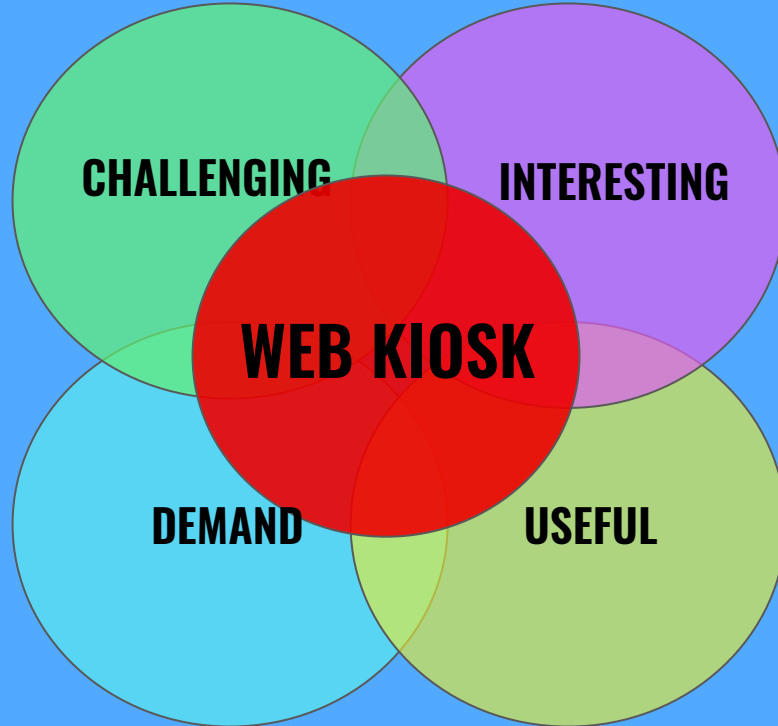
- Shopping Centres
- Airports
- Hotel Lobbies
- Work Spaces
- Other Public & Private Spaces



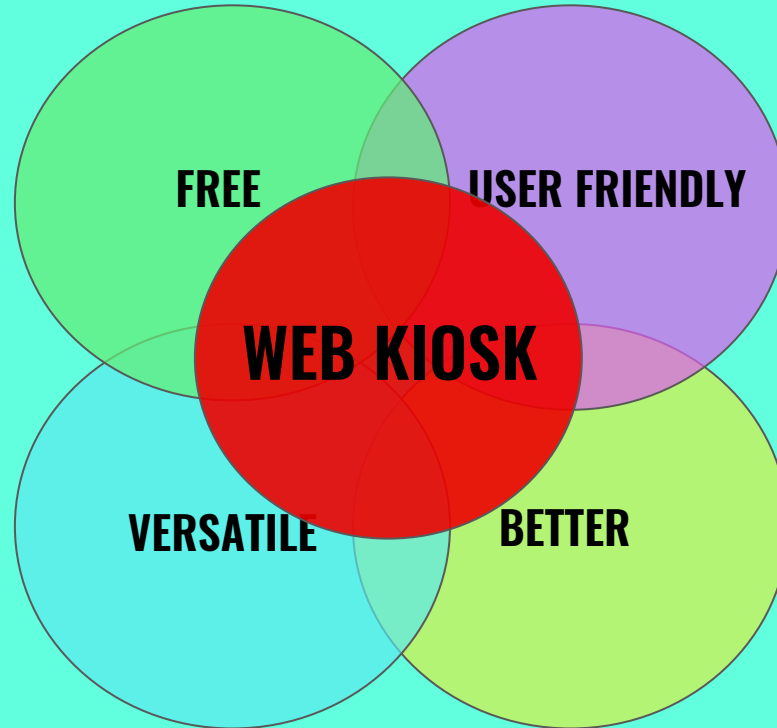
Idea



Motive



End Product



Execution



1. **Creating simple and easy to use UI using Glade**
2. **UI Interaction with Unix/Python Scripts**
3. **Python file communication with bash**
4. **Browsing websites**



chromium



Code?



- **Languages: Python 3.5.3 and Unix shell**
- **Python standard libraries utilized: os and time**
 - `time.sleep(switchRate)`
 - `os.system('./videoPlayer.sh ' + 'directory')`
- **Added libraries: GTK+ 3.22**



Working with GTK+



Connecting buttons to functions:

Define a button:

```
saveButton = Gtk.Button("Save")
```

Connect to function:

```
saveButton.connect("clicked", handler.savebutton_is_clicked)
```

Define the function:

In Handler class:

```
def savebutton_is_clicked(self, button):
```

Working with GTK+



=

box
box
box
box

LABEL	ENTRY	BTN	BTN	BTN
LABEL	ENTRY	BTN	BTN	BTN
LABEL	ENTRY	BTN	BTN	BTN
LABEL	ENTRY	BTN	BTN	BTN

Accessing widgets/containers:

```
urlbox.get_children() = [Gtk.Box, Gtk.Box, Gtk.Box, Gtk.Box]
```

```
urlbox.get_children()[0].get_children() = [Gtk.Label, Gtk.Entry, Gtk.Button, ...]
```

Get all entries texts:

```
for n in urlbox.get_children():  
    url_list.append(n.get_children()[1].get_text())
```

urlbox

Cycling browser tabs

```
while True:
    if cycle == 0:
        for n in websiteList:
            os.system('chromium-browser --incognito '+n)
            if cycle == 0:
                os.system('xdotool key "F11" &')
                cycle = 1
            time.sleep(switchRate)
    else:
        for n in range(2, len(websiteList)+1):
            os.system('xdotool key "ctrl+'+str(n)+'"')
            time.sleep(switchRate)
```

How to Use?

1. Download Source Code
2. Run the .exe File
3. Reboot



4. Configure Settings Using UI

5. Press “Run web-kiosk”

A screenshot of the 'Web Kiosk' configuration window. The window has a title bar 'Web Kiosk'. Below it is a subtitle 'Here you can choose the websites to display and define the switch and repress rates'. There is a 'URL:' text box followed by 'X', 'Up', and 'Down' buttons. Below this is an 'Add' button. Then are 'Switch Rate: 10' and 'Refresh Rate: 20', each with '-' and '+' buttons. Below these are '(None)', a folder icon, 'Load previous', and 'Save to file' buttons. A 'Run web-kiosk' button is next. The next section is 'Slideshow/Videolooping' with the subtitle 'Here you can choose a looping slideshow or video montage to play from a file or folder'. It has a 'Slideshow delay: 10' with '-' and '+' buttons. At the bottom are 'From file: (None)' with a folder icon, 'From folder: (None)' with a dropdown arrow, and a small downward arrow.

Challenges

Learning Linux

GTK & Python

Time



Future Improvements

App Support

Multiple Windows Simultaneously

Windows Compatibility

Arrangeability

Error Logs

Revenue?





