Penguin Jailed in Hacking Case

Claremont (Muddraker): Residents of East Dorm were rudely awakened Monday morning when a police battering ram suddenly knocked a hole in the dormitory's wall as part of an operation to capture a rogue penguin. “We entered at 10 AM so we could be sure everyone was asleep,” explained police captain R. Ayder. “The penguin was captured without incident.”

Interviews with the penguin’s suitemates revealed that it had been accused of computer fraud. “He was a big sushi fan,” stated F. Rosh, a first-year student. “He was always getting deliveries of fresh fish and sharing them with us. We didn’t realize that he had used an SQL injection to get it all for free.”

Another onlooker suggested that the penguin shouldn’t be punished severely. “He was lonely, all this way from the Antarctic in the Southern California heat. Who can blame him for getting fish any way he can? “
Design For Software and Beyond

Don Norman's key principles:

1. Conceptual models
2. Mapping
3. Visibility
4. Feedback
5. Affordances

Um...you can pour...
Conceptual Models

Users always bring *something* "to the table"

Images from *The Design of Everyday Things*

These don't work!

That bike looks OK to me!
Mapping

Is matching expected (spatial) relationships

Where to plug in the keyboard and mouse?
Visibility

Is making functionality *apparent*

"I used to have that awful shower controller where you pull down on the nozzle to turn it on. I had to tell every guest how to do it, and when we sold our house, we got a call from the new owners about 5 days later asking how to turn on the shower. They had been taking baths for 5 days! Unbelievable." - BL

From: www.baddesigns.com
Feedback

Providing information back to the user

Microsoft Access

Microsoft Outlook

from the UI Hall of Shame

http://homepage.mac.com/bradster/iarchitect/shame.htm
Affordances

Are the functions that form suggests…

**built-in** user's manual

Opening the XO?  Door handles
Norman's Principles

Mapping
Meeting expectations for *spatial* relationships

Visibility
Making functionality *apparent*

Feedback
Information provided by the UI to the user

Affordances
Functions suggested by an object's form
Where Do These Go Wrong?

Mapping

Visibility

Feedback

Affordances

This handle … ?

How to open this gas cap?

Set to 5 minutes?

Win NT Dialog

How to turn on this stove?
Where Do These Go Wrong?

Mapping

Visibility

Feedback

Affordances

This handle *unfastens the seat from the floor*!

How to open this gas cap?

Set to 5 minutes?

Win NT Dialog

How to turn on this stove?
User Interfaces aren't easy.

(Designers aren't users!)
1. Start with *paper prototypes*
2. Have sample users *think out loud*
3. Take notes or *(WAY better)* video them
4. Above all, *don't offer help!*
5. Analyze results, go back, fix your silly mistakes, try again
Stretch Break!

Bad Stock Photos of my Job


Tauno Talima
@tauntz

I sit in a dark room and project code straight to my face while solving complicated problems. This helps me to immerse myself in it and “feel” the code.
#BadStockPhotosOfMyJob
2:17 PM - May 4, 2018
2,220 people are talking about this

James Felton
@jimMfelton

"Hey space astronomer, where is space?"
[Space astronomer puts on his lab coat and adjusts his space safety goggles] "According to my latest science, there is space."

James William Cooper
@James_W_C

I often hold my slides and stare moodily at them. You know, instead of looking at them under the microscope that’s right in front of me. Sometimes I invite a colleague to join me.

Emma Hodcroft
@firefox66

As an evolutionary biologist, you have *no idea* how hard it is to find tweezers small enough to grab bits of DNA (which is the main part of our job, obvs)
#BadStockPhotosOfMyJob
3:42 PM - May 4, 2018
Desktop apps: tkinter

**editor.py**

```python
from tkinter import Tk

def main():
    app = Tk()
    ## Do some stuff…
    app.mainloop()
```

Now we’ve got a Graphical User Interface!

Use the tkinter library

Create a top-level application

Run your application
from tkinter import TK, Frame

class Editor(Frame):
    def __init__(self, parent):
        Frame.__init__(self, parent)
        self.parent = parent
        self.init_window()
    def init_window(self):
        self.parent.title("CS5 Editor")
        self.pack(fill=BOTH, expand=1)

def main():
    app = Tk()
    editor = Editor(app)
    app.mainloop()
class Editor(Frame):
    def init_window(self):
        self.parent.title("CS5 Editor")
        self.pack(fill=BOTH, expand=1)

        menu = Menu(self.parent)
        self.parent.config(menu=menu)

        file_menu = Menu(menu)
        file_menu.add_command(label="Open", command=self.open)

        menu.add_cascade(label="File", menu=file_menu)
On your worksheet...

- Add File->New and File->Save
- Add a second Menu of your choice

```python
class Editor(Frame):
    def init_window(self):
        self.parent.title("CS5 Editor")
        self.pack(fill=BOTH, expand=1)

        menu = Menu(self.parent)
        self.parent.config(menu=menu)

        file_menu = Menu(menu)
        file_menu.add_command(label="Open", command=self.open)

        menu.add_cascade(label="File", menu=file_menu)
```

Other Widgets...

**editor.py**

```python
def init_window(self):
    // Skipped for space...
    menu.add_cascade(label="File", menu=file_menu)

    self.text = Text(self)
    self.text.pack(fill=BOTH, expand=1)

def open_file(self):
    self.fname = filedialog.askopenfilename()
    self.text.insert(END, open(self.fname).read())

def save_file(self):
    open(self.name, "w").write(self.text.get("1.0", END))

def client_exit(self):
    self.parent.quit()
```
tkinter demo!
from tkinter import *
from tkinter import filedialog

Adding a top-level comment in class!

class Editor(Frame):
    def __init__(self, parent):
        Frame.__init__(self, parent)
        self.parent = parent
        self.init_window()
        self.filename = None

    def init_window(self):
        self.parent.title("CS5 Editor")
        self.pack(fill=BOTH, expand=1)

        menu = Menu(self.parent)
        self.parent.config(menu=menu)

        file_menu = Menu(menu)
        file_menu.add_command(label="Open", command=self.open_file)
        file_menu.add_command(label="Save", command=self.save_file)
        file_menu.add_command(label="Exit", command=self.client_exit)

        menu.add_cascade(label="File", menu=file_menu)

        self.text = Text(self)
        self.text.pack(fill=BOTH, expand=1)

    def open_file(self):
        fname = filedialog.askopenfilename()
        self.filename = fname
        text_content = open(fname).read()
        self.text.insert(END, text_content)
        self.has_changes = False

    def save_file(self):
        open(self.filename, "w").write(self.text.get("1.0", END))

    def client_exit(self):
        self.parent.quit()

def main():
    app = Tk()
    editor = Editor(app)
    app.mainloop()

if __name__ == "__main__":
    main()
Web apps: Flask

Functions and decorators

Use the flask library

Create a top-level application

Map functions to paths

```python
from flask import Flask

app = Flask(__name__)

@app.route('/

def hello_world():
    return 'hello, world!'"
**Flask Variables**

```python
cs5_site.py

@app.route('/homework/<int:hw>')
def display_assignment(hw):
    return 'homework {}!'.format(hw)
```

*Int variable in the path*

*Passed as parameter*
Flask Templates

cs5_site.py

```python
@app.route('/homework/<int:hw>')
def display_assignment(hw):
    return render_template('homework.html', hw=hw)
```

templates/homework.html

```html
<!DOCTYPE HTML>
<title>CS5 Homework {{ hw }}</title>
<h1>Here is what you should do for Homework {{ hw }}!!!</h1>
```
flask demo!
from flask import Flask, render_template

app = Flask(__name__)

@app.route('/')
def index():
    return "Welcome to the CS5 Homepage!

@app.route('/homework/<int:hw>')
def display_assignment(hw):
    return render_template('homework.html', hw=hw)

@app.route('/hello')
def hello_world():
    return 'Hello, World!'
Here is what you should do for Homework {{{hw}}}!!

- Download some files
- Write some code
- Submit to Gradescope

{% if hw == 0 %}
I hope you enjoyed your first CS5 homework!
{% else %}
One more homework down (and only {{{12-hw}}} left to go)!
{% endif %}