

# CSE 410 - Human Computer Interaction

## The **desktop** and JavaFX (project 2)

### Overview

Desktop applications play an important role in handling applications that are resource intensive, need to be always available, have security considerations, or need to access hardware. There are many different formats for coding desktop applications, but for this lab we will be using the JavaFX library. Along with experience with this library, we'll explore packaging a java application as a native executable, CSS styling for the desktop, using the SceneBuilder tool, and the vCard format for exchanging contact information.

### Resources and Lab Setup

For this lab, you will need to have Java 8 installed, and it's recommended that you have Eclipse or a similar suitable IDE on your development platform. This lab was tested with Eclipse Oxygen and Java 1.8.0.144. You will also need to install the [SceneBuilder](#) tool for Java 8. There are several additional .jar files that you will need for various aspects of the project, which can all be download in a [zip file](#) from my website.

### Requirements

The first step in the lab is to [complete a tutorial](#) on building a basic JavaFX address book application. Once you have completed the tutorial, we'll take the now functional java application and make some additional modifications!

#### *Extend the format handling for birthdays*

Use the work that we did in class coming up with different ways that dates can be entered to create a more flexible input for dates. If you use text input, it needs to handle at least 5 different formats of input dates, and disambiguate where necessary.

#### *Add some additional data elements*

In order to get a feel for adding additional UI components, and then taking the data through the serialization step, add the following fields to the person record.

1. A photo of the contact. If no photo is available (the default state), a placeholder image should be displayed.
2. An e-mail address for the contact. The input should verify that the format is  
[ [a-z] | [ . \_ - ] ] \* @ [ [a-z] | [ . ] ] \*
3. A phone number for the contact. This should take all formats and normalize to (123) 456-7890.
4. A URL for the contact. The URL should be launchable from app.

If any of the inputs are invalid, an appropriate guidance message should be provided.

*Add a dash of (your own unique) style*

Craft a new stylesheet that applies your own look and feel to the application to replace the standard “dark” theme from the tutorial. This can be whatever you like, but needs to be consistent, and there is a subjective aesthetic component of the grade, so no blinking red on blue text with emoji flood fill backgrounds, please.

*Provide context sensitive guidance*

Error handling is done in the tutorial code via popup message, and does not provide much guidance on how to correct problems in data entry. Change the error handling to make it context aware (the inputs in question should be visually indicated, and guidance text should be in proximity to those input fields).

*Drag and Drop*

Add the ability to drag and drop a vCard file onto your application to add a new contact. You may want to check out some tutorials on [drag and drop](#) and the [vCard format](#).

*Export a vCard file*

Finally, add the ability to export a contact from your application to a vCard file. Where you put the control and how you signify that functionality is up to you - use what you have learned so far to put it in a good place!

## Submission

A windows .exe file should be submitted via ulearns.

## Scoring

Scoring for the lab will be as follows

- (9 Points) Working application installs from .exe file
- (3 points) Birthday input format handles at least 5 different formats
- (3 points) Guidance text is context sensitive
- (3 points) Photo can be added via file chooser
- (3 points) Photo displays default image if not specified
- (3 points) Error handling for non-image files selected
- (3 points) email address can be added
- (3 points) email address validation works correctly
- (3 points) phone number can be added
- (3 points) phone number formats correctly
- (3 points) phone number is validated to ensure correct number of digits
- (3 points) URL can be added
- (3 points) URL can be launched from app
- (3 points) New stylesheet created
- (3 points) Stylesheet consistently applied throughout app
- (6 points) Drag and Drop import of vCards works

- (3 points) Export of vCards works
- (5 points) Subjective aesthetics