

**Class Website:** <http://www.ecs.csun.edu/~cov/comp585f15>, few hardcopy handouts, most info posted to class website as pdfs.

### Online Texts

- Creating a GUI with JFC/Swing, aka Swing Tutorial, online at <http://docs.oracle.com/javase/tutorial/uiswing/>

### Recommended Texts

- Matthew MacDonald, Pro WPF in C# 2010: Windows Presentation Foundation in .NET 4, Apress, 2010  
Newer version: Pro WPF 4.5 in C#, Apress, 2012
- Liberty and MacDonald, "Learning C# 3.0", O'Reilly, 3<sup>rd</sup> edition, 2008. (basic level)
- Griffiths, Adams, and Liberty, "Programming C# 4.0", O'Reilly, 6<sup>th</sup> edition, 2010. (advanced level)

### Goals

This course discusses the role of the graphical user interface (GUI) within the software application design and implementation process. Projects emphasize GUI building for traditional desktop environments, with some discussion of the migration of desktop apps to the web browser in the form of RIAs (Rich Internet Applications). Computer graphics topics will be included as they relate to GUIs, but in-depth computer graphics is covered in separate courses. We will study and discuss representative application programming interfaces (APIs) for development of GUIs using existing popular languages (Java and C#). The design of GUIs requires attention to both presentation (how to layout elements of the GUI in two dimensions that the user perceives as effective, complete, and easy to understand and use) and behavior (how to use programming techniques to arrange for the GUI's correct behavior). Specifically, the Java Swing API is explored in depth, and is used as the basis for several programming projects in the first half of the course. Next, the C# language and the Windows Forms (WinForms) and Windows Presentation Foundation (WPF) namespaces provided by the Microsoft .NET Framework are explored in depth, and are used as the basis for programming projects in the second half of the course. Finally, we will look at some RIA platforms such as Java FX in overview. In-depth web application engineering is covered in separate courses.

Although there is no lab, this is not a whiteboard course for discussing GUI concepts at an abstract level. Students are expected to acquire practical programming skill and mastery of an important subset of the discussed APIs. Competent programming skill at the level of COMP 282 is a prerequisite to successfully complete the projects for this course. If you have no knowledge of Java, you will need to catch up quickly on your own, or postpone taking the course until you do. No previous experience in C# is required, basic language details will be introduced in lecture, but you will need to practice and acquire proficiency on your own outside of class.

### Schedule (Tentative)

- Concepts, What Problem are we trying to solve.
- Review of Special Java Programming Topics and Overview of Java Swing and Java AWT (2 weeks)
- In-depth Study of the Java Swing Packages and APIs (6 weeks)
- Introduction to C# and Microsoft .NET Windows Forms and WPF (5 weeks)
- Miscellaneous Topics (RIAs, etc.)

### Grading

- Midterm 30%
- Final 30%
- Programming projects 40% (4 or 5)

Plus/minus grading will be used. Exams will contain a mixture of design, coding, tracing, and conceptual questions.

### Projects

Projects will be organized as 1 or 2 small Java "warm-up" projects, 1 or 2 larger Java project, 1 or 2 small C#/WPF "warm-up" projects, 1 larger C#/WPF project. I am open to the idea of students proposing projects for class credit that are of individual interest. The discussion of text editing in the Java Swing tutorial is probably the starting point for one Java project: a simple text editor. Other possible "larger" projects are GUI front ends for database back ends, using specialized APIs such as JDBC to connect from the application to the database, and extract data and populate the GUI.

For Java projects, use the latest version of the JDK. It is a requirement for programming projects in this class that you build your projects programmatically rather than automatically generating them with a GUI builder. For C#, use the latest version of the command prompt compiler tools and libraries available as part of the Microsoft SDK (minimalist solution) or the Microsoft Visual Studio .NET application. The SDK is a free download available from <http://msdn.microsoft.com>. Visual Studio is a commercial application. The CECS college participates in a Microsoft MSDN program that allows students to download many software applications including Visual Studio.

For C# projects using the .NET libraries (namespaces), GUIs built with the WinForms libraries can be built without an IDE. In contrast, projects built with the Windows Presentation Framework (WPF) will require Visual Studio. WPF projects have complex build procedures and use several preprocessors. Microsoft doesn't provide any special support for building these apps from the command line.

### **Academic Honesty**

Academic honesty is expected in this course. Exams and projects must be completed according to the guidelines established by the instructor for that assignment. Penalties for plagiarism and other forms of cheating (including participation in rent-a-coder type services as either a buyer or a seller) is grounds for receiving an F on the exam or project in question, and possibly an F for the entire course. Violations may be reported to the Department Chair, College Dean, and the Office of Student Affairs.

### **Topics**

- Guidelines for GUI Design
- APIs for GUI Programming
  - Sun JDK (specifically the AWT and Swing packages)
  - Microsoft .NET (Windows Forms and WPF, accessed from C# API)
- Event-Driven Programming
- Relevant Java Language Features
  - Nested Classes
  - Inheritance, Interfaces, Adapters
  - Threads
  - Events and Listeners
- GUI Building Blocks
  - Containers, Components, Layout Managers
- Other General Java Topics
  - Internationalization, Standard Formatting, and Utilities
  
- Relevant C# Language Features
  - Events and Delegates
  - Properties
  - Inheritance and Virtual Functions
  - class v. struct types (references v. values)
  - Arrays (rectangular vs. jagged)
  - Lambda Expressions
- GUI Building in Windows Presentation Foundation (WPF) with C#
  - Precursors: Win32 library (many books by Petzold), MFC (Microsoft Foundation Classes and C++)
  - .NET and Windows Forms (development is frozen ("maintenance mode"), still in commercial use)
  - .NET and WPF: C# and XAML (eXtended Application Markup Language), both standalone (desktop) apps, and XBAP (XAML Browser App) apps (like a desktop app but hosted in a browser)
- Advanced GUI Topics
  - Trees
  - Tables
  - Images
  - Custom GUI Components
  - Other Possible Topics
    - Other APIs: Qt (uses signals and slots), GTK+, others
    - Eclipse and the Standard Widget Toolkit (SWT)
    - Mobile Platforms
    - Touch Screen APIs (multi-touch)
    - RIAs (Java FX, Microsoft Silverlight (deprecated), Flash/Flex, HTML5/JavaScript/GWT)