ITP 368: Programming Graphical User Interfaces
Units: 4
2020 Fall
Instructor: Dr. Barrett Koster
Office: online
Office Hours: MW 10-11, TH 1-3

section: 31807
day+time: M W 12:00pm - 1:50pm
Location: online

Contact Info: All general course/assignments questions should be asked on Piazza (every student will receive an invitation at the start of the semester).

Other questions should be asked via email: bkoster@usc.edu
(General timeline for replying to emails is within 24 hours)

Teaching Assistant:
Yiqian
Anthony
more info on Piazza

IT Help: Provided by Viterbi IT
Hours of Service: 8am–5pm M-F
Walk-in: DRB 205 (or not)
Contact Info: (213) 740-0517
Email: engrhelp@usc.edu

Course Description
Programming applications with dynamic graphical user interfaces. Topics include events, controls, resources, data bindings, styles, and user experience. Students will learn industry best-practice approaches for software project design using design patterns and how to understand and apply fundamental UI design principles while programming desktop UI components. Students will go through the process of building desktop applications from start to finish using Java. You will learn how to leverage your Java programming knowledge and explore JavaFX libraries to design graphical interfaces, simple animations, and mini-games. Students will also consider design aspects such as localization and internationalization, as well as accessibility for end users. Students enrolled in this course should have strong object-oriented programming skills and be able to apply their knowledge of data structures towards problem solving.

Note for online classes
This semester we are online, all remote. Students in compatible time zones are expected to join the zoom class meetings live and submit labs same-day (although attendance is not *required*). Students for whom live class is the middle of the night (or if you have some other difficulty with class) may watch class recordings of class -- all classes will be recorded -- and submit labs offline. I am proud to be having this class around the world, unstopped by the virus. We will have TA and/or my office hours to accommodate. If there are other needs, let me know. Welcome.

Learning Objectives
- Strengthen programming and software design skills

- Use APIs and tools to build and design GUI applications: animations, games, and user interfaces
- Identify and use some software design patterns.
- Understand and apply fundamental UI design principles in GUI programs.
- Understand and implement accessibility considerations in your software.
- Understand why we need localization and internationalization of programs and how to set up applications to easily be localized
- Implement a self-directed demo-quality project and present it

Prerequisite(s): ITP 265 and/or CSCI 103, knowledge of Java

Course Notes
Format: This course will make use of several tools for content and assignments including Blackboard and an online class discussion forum (Piazza). Lecture notes and any supplemental course content will be posted to Blackboard for use by all students. Any and all announcements for the course will be posted to the class discussion forum. All assignments will be posted to Blackboard and will be submitted through Blackboard.

Required Readings and Supplementary Materials
You might find these resources helpful.
- Head First Design Patterns by Eric Freeman, Elisabeth Robson, Bert Bates, and Kathy Sierra
- Mastering JavaFX 10 by Sergey Grinev
- Core Java Volume I and Volume II by Cay Horstman
- Learn JavaFX8: Building User Experience and Interfaces with Java 8 by Kishori Sharan
You can get access to these through an ACM student membership for $19.

Course Structure
The class meets for one hour and 50 minutes twice a week for a total of 3 hours and 40 minutes. Two exams and possibly several in-class quizzes will be given. Programming assignments or projects will be assigned to be completed outside of class time. Access to a laptop computer during class is required. ITP does have a laptop loaner policy for students enrolled who do not have a personal laptop.

Grading Breakdown

Item	% of Grade
Assignments (weighted proportionally)	40
Tests (2 total)	30
Final Project	25
labs	5
Total	**100**

Grading Scale
Course final grades will be determined using the following scale
A   95-100
A-   90-94
B+   87-89
B   83-86
B-   80-82
C+   77-79
C   73-76
C-   70-72
D+   67-69
D   63-66

D-   60-62
F   59 and below

If you are taking the class with a grade of P/NP, you must earn a grade of 70% or higher in order to receive a P.

Programming Assignment Policies

Programming assignments will generally be due one week after they are assigned and should be completed individually. All code should be submitted on Blackboard (unless otherwise stated) and must compile.

Homework

The assignments will be posted on the Blackboard Assignments. Each assignment will include instructions, requirements, point breakdown, a due date, and a link for electronic submission. Assignments must be submitted using this link.

It is the student's responsibility to submit assignments **on or before** the due date. Assignments turned in up to 24 hours late will have 15% of the total points deducted from the graded score. Assignments turned in 24-48 hours late will have 30% of the total points deducted from the graded score. Assignments turned in 48-72 hours will have 50% of the total points deducted from the graded score. After three days, submissions will **not** be accepted, and will result in a score of 0 (zero). Each student will be allowed **ONE** 24-hour late assignment for "free", which may not be used on final project.

If you have a valid excuse for turning in an assignment late, send me email with explanation/documentation and also (or if approved) submit to Blackboard a summary of your request as a 'first draft' of your homework. I will add a note on Bb with my decision and typically give a score of -1, which lets the TAs know this is pending.   You can then submit your actual assignment later as a 2nd version for a real score. Note: the critical part is the submission to Bb -- I can't leave a note for the TA to accept your work late unless there is a submission there for me to put the note on.

Students are required to keep a copy of all of their assignments. Frequent backups to an external drive or to the cloud is strongly recommended. ITP is not responsible for any work lost.

Grading Timeline

Assignments will be graded, and students will receive feedback within two weeks after submission.

Grading Issues

Students will have two weeks after graded feedback is given to contest scores (e.g. assignments, quizzes, exam, and project). After two weeks, scores will not be changed.

Submissions should comply with assignment, but there are a lot of features to JavaFX that we will not have time to cover, so creativity is welcome. If you provide additional functionality or design to your submission, it should demonstrate competence with the feature and be well documented. I want you to show you have control of the medium, not just that you are jumping through the hoops of the assignment.

Additional Policies

General

No make-up exams or tests (except for documented medical or family emergencies) will be offered.

Attendance may be taken during lecture sessions electronically, verbally, or via a roster passed around the room. Do not sign in for another student; doing so is an academic integrity violation. Attendance is not mandatory, but students are responsible for any announcements made during lecture time and understanding material covered in class. Student work will be graded on the assumption that they have mastered material from class.

Do not reproduce, distribute, or post any lecture material, assignments, or exams publicly without my written consent. Students may take notes and make copies of course materials for their own use. Students may not post my course materials on sites such as CourseHero. Doing so is a copyright violation and an academic integrity violation that will be dealt with accordingly.

ITP offers open lab use for all students enrolled in ITP classes. These open labs are held beginning the second week of classes through the last week of classes. Hours are at https://itp.usc.edu/current-students/open-lab-schedule/. In addition, ITP has a laptop loaner program for students who may need temporary use of a laptop in order to complete an assignment.

Late Add

Per university policy, students are allowed to add the course until the end of week three. Any students wishing to add the course should plan on attending the course from the beginning of the semester. Upon adding the course after week 1, the student should email the instructor **immediately** to make sure there is a plan for completion of work and learning missed materials. Any missed work is required to be completed and submitted according to the schedule provided by the instructor. Generally, late arriving students should be caught up by week 5.

Academic Integrity

USC seeks to maintain an optimal learning environment. General principles of academic honesty include the concept of respect for the intellectual property of others, the expectation that individual work will be submitted unless otherwise allowed by an instructor, and the obligations both to protect one's own academic work from misuse by others as well as to avoid using another's work as one's own. All students are expected to understand and abide by these principles.

Plagiarism – presenting someone else's work or ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in **SCampus** in Part B, Section 11, "Behavior Violating University Standards" https://policy.usc.edu/scampus-part-b/ . Other forms of academic dishonesty are equally unacceptable. See additional information in **SCampus** and university policies on scientific misconduct, http://policy.usc.edu/scientific-misconduct.
Academic integrity tutorials can be found at https://libraries.usc.edu/research/reference-tutorials

Examples of behavior violating University standards:
- The submission of material authored by another person but represented as the student's own work, whether that material is paraphrased or copied in verbatim or near-verbatim form.
- Obtaining for oneself or providing for another person a solution to homework, a project or other assignments, or a copy of an exam or exam key without the knowledge and expressed consent of the instructor.
- Unauthorized collaboration on a project, homework, or other assignment.

If the instructor, a grader, or a lab assistant **suspects** you of academic dishonesty, it has to be reported to SJACS (https://sjacs.usc.edu). Do not share assignments with any other people. Do not submit another person's work as your own. Do not look at other students' papers during exams.

Do not leave the room during an exam without permission. **Do not cheat! As Trojans, we are faithful, scholarly, skillful, courageous, and ambitious.**

Viterbi Honor Code

Engineering enables and empowers our ambitions and is integral to our identities. In the Viterbi community, accountability is reflected in all our endeavors.
Engineering+ Integrity.
Engineering+ Responsibility.
Engineering+ Community.
Think good. Do better. Be great.
These are the pillars we stand upon as we address the challenges of society and enrich lives.

** For final exam presentations, students will present in 10-15 minute slots, with times available during their final exam period, but with additional time slots available and scheduled via a Doodle poll.

Statement on Academic Conduct and Support Systems

Academic Conduct
Plagiarism – presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in *SCampus* in Part B, Section 11, "Behavior Violating University Standards" https://policy.usc.edu/scampus-part-b/. Other forms of academic dishonesty are equally unacceptable. See additional information in *SCampus* and university policies on scientific misconduct, http://policy.usc.edu/scientific-misconduct.

Support Systems
Student Counseling Services (SCS) - (213) 740-7711 – 24/7 on call
Free and confidential mental health treatment for students, including short-term psychotherapy, group counseling, stress fitness workshops, and crisis intervention. https://engemannshc.usc.edu/counseling/

National Suicide Prevention Lifeline - 1-800-273-8255
Provides free and confidential emotional support to people in suicidal crisis or emotional distress 24 hours a day, 7 days a week. http://www.suicidepreventionlifeline.org

Relationship and Sexual Violence Prevention Services (RSVP) - (213) 740-4900 - 24/7 on call
Free and confidential therapy services, workshops, and training for situations related to gender-based harm.
https://engemannshc.usc.edu/rsvp/

Sexual Assault Resource Center
For more information about how to get help or help a survivor, rights, reporting options, and additional resources, visit the website: http://sarc.usc.edu/

Office of Equity and Diversity (OED)/Title IX Compliance – (213) 740-5086
Works with faculty, staff, visitors, applicants, and students around issues of protected class.
https://equity.usc.edu/

Bias Assessment Response and Support
Incidents of bias, hate crimes and microaggressions need to be reported allowing for appropriate investigation and response. https://studentaffairs.usc.edu/bias-assessment-response-support/

The Office of Disability Services and Programs
Provides certification for students with disabilities and helps arrange relevant accommodations. http://dsp.usc.edu

Student Support and Advocacy – (213) 821-4710
Assists students and families in resolving complex issues adversely affecting their success as a student EX: personal, financial, and academic. https://studentaffairs.usc.edu/ssa/

Diversity at USC
Information on events, programs and training, the Diversity Task Force (including representatives for each school), chronology, participation, and various resources for students. https://diversity.usc.edu/

USC Emergency Information
Provides safety and other updates, including ways in which instruction will be continued if an officially declared emergency makes travel to campus infeasible, http://emergency.usc.edu

USC Department of Public Safety – 213-740-4321 (UPC) and 323-442-1000 (HSC) for 24-hour emergency assistance or to report a crime.
Provides overall safety to USC community. http://dps.usc.edu

Schedule for ITP-368 GUIs

2020 Fall

week	M	class agenda	W	class agenda (HW due Friday midnight)
1	Aug 17	slides: Course Information , Windows/JavaFX basics , syllabus aux: Hello.java , HelloFX.java , MyCircleApp.java , OneLine.java , lab: install IDE, run HelloWorld, run HelloFX	Aug 19	slides: Java Basics , Drawing Basics aux: DrawStuff.java , Dots.java lab: groups make images due: HW00 Anything
2	Aug 24	slides: Buttons and Events . aux: AButton.java , LabelDemo.java , AButton2.java , lab: make buttons to change color, add boxes, Tic Tac Toe board.	Aug 26	slides: Layouts aux: MoreButtons.java and , ScrollPane demo LayoutDemo.java (build this live, use fn.s) and RobotWorld.java lab: reproduce given layout (picture provided) due: HW01 Art
3	Aug 31	look at the Art programs slides: CSS intro , Key Events , (Objects 1) , aux: ColorBox.java and ColorBoxDemo.java , Key Demo Widgets , scene swapping, window sizing/resizing, and widgets live: robot? lab: Dice panels with roll buttons.	Sep 2	slides: Collections . Files . aux: Dice.java , DiceDemo.java , ListPlay.java , NumberBox.java , Exceptional.java , FileReadDemo.java , XX robot stuff XX, lab: add day events to calendar from file due: Converter for unit conversion.
4	Sep 7	Labor Day no class	Sep 9	look at some calculator programs slides: Mouse1 . (location from click) aux: MouseHello.java , TicTacToe1.java , TicTacToe2.java , TicTacToe3.java , live: Mancala.java , starter code (maybe) : Manc0.java , Pit.java lab: Lights Out
5	Sep 14	slides: Mouse2 (dragging and selecting). aux: Boxer2.java , Boxer3.java , Boxer4.java , Boxer5.java , Boxer5k.java , Point.java , Complex.java , OvalFiles.java live: zot (Mandelbrot set viewer) lab: Elf. due: Deal or No Deal	Sep 16	topic: another day on mouse program structure, tricks, fn.s, classes. make frame for Paint
6	Sep 21	topics: Design for Access , access demo , Kendra's talk , lab: YouBet.java	Sep 23	slides: Schneiderman's 8 principles , Gestalt , What Users Do , Localization lab: rate programs on interface criteria due: Paint
7	Sep 28	Review for test 1 .	Sep 30	test1

8	Oct 5	slides: simulation , live: PlainOldLife.java , with Cell.java , lab : Fish Shark Life etc. Don't be too ambitious. Build more than Show.	Oct 7	project - GO! Project general, demos slides: MVC design , Tic Tac Toe w 2 IFs lab: do core of Checkers
9	Oct 12	slides: Animation (Transitions - figure these out!) live: students do transition, demo to each other, but here is one. Ani3.java (TimeLine demo) lab: fix Mancala starter to make Mancala smooth ... both use Pit.java , :Point.java , Stone.java due: Neighborhood	Oct 14	slides: Animation2 aux: Ani2.java uses Ball2.java lab: BrickOut (goal) uses Ball and Brick
10	Oct 19	slides: synchronization , could do with this: Concurrency live: lots of demos, shared printer	Oct 21	live: readers and writers (starter version and solution lab: share TWO printes due: Maxwell
11	Oct 26	slides: a-sync communication , live / lab: tturn Chatter.java into tic tac toe on 2 computers	Oct 28	Sleeping Barber starter code
12	Nov 2	slides: Audio basics (and video). live: DemoClip.java , DemoMediaPlayer.java , DemoRecord.java , VideoDemo.java , Webber.java , lab: make sound board and my target uses SoundPane.java	Nov 4	slides: SoundGraph with code : Grapher.java lab: turn Grapher into 2 panels with full wave and selection ?
13	Nov 9	slides: Review 2 , Deployment ,	Nov 11	project demos and/or work perspectives
14		other graphical systems ...		other systems ...
15		(insert this as week 13?) Generics . live: generic demo lab: create generic sorts		Properties ? OO Design ? survey other GUI environments!
exam	Fri Nov 20	11AM-1PM project demos		

--module-path "/Program Files/JetBrains/javafx-sdk-14.0.2.1/lib" --add-modules javafx.controls,javafx.fxml,javafx.media