1 Course Staff

<table>
<thead>
<tr>
<th>Kelly Wilkerson</th>
<th>Brett Ussher</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="mailto:kwilker2@asu.edu">kwilker2@asu.edu</a></td>
<td><a href="mailto:bussh@asu.edu">bussh@asu.edu</a></td>
</tr>
</tbody>
</table>

Office Hours will be Tuesdays 11:30am-1:30pm in Prof. Wilkerson’s office (M1-39), unless specified otherwise on Blackboard. Please check Blackboard for up-to-date information about office hours, as they might change depending on the needs of the course.

2 Course Description

Mobile devices take the term “personal computing” to the extreme. As the computer that we carry with us at all times, our mobile device becomes our news station, our map and compass, our camera, and our conduit to the rest of society. However, we use this computer in different ways than we use any other computer; our interaction with it is frequent, frequently interrupted, and frequently taken for granted.

This course will introduce students to application development for mobile devices. Students will learn about the various constraints facing mobile application designers, both with respect to hardware and with respect to user expectation. Students will also learn how to address these constraints with techniques in implementation, software design, and user-interaction design. Additionally, students will also learn about concepts at the core of modern mobile computing, such as software and data distribution models and location awareness.

The course focuses on using the iPhone OS as the development platform, but the concepts covered in the course are platform agnostic. As such, students will be introduced to the Objective-C programming language, the XCode programming environment, and the iPhone SDK and APIs.

This course will focus heavily on in-class participation, out-of-class assignments, and programming projects.

3 Major Topics Covered

- User-interaction design and requirements design
- Graphical User Interfaces and Event-Driven Programming
- Advanced Object-Oriented Programming
- Robust design and programming for user constraints (application interruption, application responsiveness, partial user engagement)
- Robust design and programming for device constraints (power consumption, screen size, network connectivity, memory limitations)
- Data distribution: distribution types, basic parsing, distribution security.
- Location awareness, messaging, and other connections between the device and the outside world.

4 Topics We Might Cover

- Rendering graphics
- Animation
- Image recognition and processing
- Interface polishing
5 Lecture Times

Lectures are held in the new mezzanine space BYENG M1-11. Attendance will not be taken, but material presented in lecture will be critical to completing homework assignments and fulfilling course goals. Additionally, there may be in-class activity assignments that you will need to make up if you are absent.

6 Prerequisites

The official prerequisite for this course is CSE 310. You are expected to have a good understanding of Object-Oriented programming and C or C++ programming. You should have experience writing moderately-sized programs, to prepare you for writing larger programs in this course.

If you don’t meet the official prerequisites but are admitted to the course because you have equivalent experience, it is your responsibility to make sure you understand the necessary background material.

7 Textbooks

A majority of the relevant material for this course will come from online resources, API documentation, and notes written specifically for this course.

8 Grading

Your performance will be assessed through homework assignments and course participation.

Your letter grade will be decided by a scale no more strict than the following:

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>90-100%</td>
</tr>
<tr>
<td>B</td>
<td>80-89%</td>
</tr>
<tr>
<td>C</td>
<td>70-79%</td>
</tr>
<tr>
<td>D</td>
<td>60-69%</td>
</tr>
<tr>
<td>E</td>
<td>less than 60%</td>
</tr>
</tbody>
</table>

The grades A+, B+, and C+ are possible, but the required percentage will not be decided until final grade calculations.

9 Assignments

Assignments in this course will be programming projects, reading assignments, and some in-class activities. Consecutive assignments may be organized to form a larger programming project. Assignments will be posted to the course website, and due dates will be added to the course calendar.

We urge you to begin assignments as soon as possible to avoid the bottleneck that inevitably occurs as due dates approach. You are responsible for completing your assignments on time, even in the face of server problems.

10 Re-grading

We recognize that you may not be satisfied with a particular grade, and we have adopted a course-wide policy for dealing with these issues:

Any questions, corrections, or appeals on grades of assignments must be done in writing within one week after it has been returned to the class. State the problem and the rationale for any change in your grade in your appeal. For assignments, contact the staff member that who graded it.

Questions, corrections, or appeals will not be accepted without a copy of the graded item and a written/email appeal (this is so we can walk away with it and have a reminder of what we discussed with you). The item and appeal will be taken and evaluated. Any appeal will result in an evaluation of the entire assignment, and may result in a score that is higher or lower than your original score.
11 Collaboration

Unless otherwise instructed, you are allowed and encouraged to discuss assignments with other students and exchange ideas about how to solve them. Programming assignments will be completed in small groups; reading assignments are to be completed individually. All discussions about assignments not within a group should be limited to ideas about how to approach the assignment and should never stray to the code that to be written. When in doubt, follow this rule of thumb: Discussions should in English (or the natural language of your choice) not in code.

We will use software to detect instances of cheating, and we have no problem reporting them. All instances of cheating will be handled by the Dean’s office according to ASU Student Academic Integrity Policy and the USI 10401: Student Code of Conduct and Student Disciplinary Procedures.

12 Disclaimer

I reserve the right to revise this syllabus as necessary. Any changes to the syllabus will be announced via the course website.