SI 669 - Developing the Mobile Experience

Course Syllabus

Fall 2019 Semester
University of Michigan School of Information

This syllabus is all potentially subject to change over the course of the semester. Any substantial changes to it will also be announced.

Mental Health and Well-Being at the University of Michigan

Instructor

Instructor: Mark W. Newman, Associate Professor
Email: mwnewman@umich.edu (please see Communication section before contacting directly)
Office Hours: TBD.

Course Description

This course will introduce you to developing mobile applications that take advantage of uniquely mobile features, such as geolocation, cameras, and notifications. In the 2019 offering of the course, you will gain a solid understanding of building applications using React Native and Firebase, as well as how to use standard testing and debugging tools to find and fix software bugs. You will also receive a general overview of other mobile app development approaches.

The course is intended to be programming-intensive. In other words, the emphasis will be more on the "Developing" than the "Experience", though we will talk about what makes the "Mobile Experience" different from other kinds of interactive experiences, and use that knowledge in the things that we build.

Learning goals

- Competency: **Build a Minimally Viable Prototype** of a complete mobile app, including native components (e.g., camera, geolocation), UI, and a backend using React Native and Firebase.
- Competency: Use state of the art **tools for developing, testing, and monitoring** mobile applications.
- Competency: Argue effectively for the **strengths and weaknesses of different mobile development approaches**, including responsive web design, platform independent frameworks, and native components.
- Literacy: Understand the **mobile app ecosystem**, including Native APIs, App Stores, beta testing tools, push notifications, and mobile app UIs.

Required prerequisites

506, 507, 539 (or waivers thereof)

Beneficial prior courses and experience

664, 582, Additional experience with JavaScript beyond 539

Exceptions to prerequisites

The purpose of the prerequisites is to make sure that you are ready to dive into a new development platform with enough confidence and programming proficiency to be able to pick up new ideas and approaches quickly, and be able to troubleshoot problems you encounter, including searching for solutions online and in platform documentation. The primary language used in the course will be JavaScript, so you are expected to have or be able to rapidly acquire a good grasp of how to read and write JavaScript programs. You will need to have a basic understanding of how HTML/XML, CSS, and JavaScript work together, and will need to know how to use HTML/XML templates to render dynamic content. The specifics of the templates used in React Native will be covered in class, but the general concept of templating is something you are expected to understand coming in.

If you haven't taken the prerequisites, but feel that you have the background to succeed in this course, I am open to waiving the prereqs. Be advised, though, that the teaching team (i.e., I) will not have the capacity to fill in gaps in the expected prerequisite knowledge and skills.

Textbook and Required Resources

There is no textbook for this course. As far as I know there does not exist a textbook (or, more generally, a book) that introduces mobile app development (in React Native or any other platform) at a level that would be appropriate for intermediate programmers. I am striving to develop the course Lecture Notes in a way that is approximately as self-contained and complete as a textbook would be. Perhaps some day these will evolve into a textbook that can be shared with others, but for now they are a work in progress. The Lecture Notes sometimes refer to external resources--all of these will be freely available on the web.

You will need to have access to an Android or iOS smartphone. This does not need to be your own personal phone, as long as you can gain access to the phone for testing. For in-class exercises requiring access to a phone you can partner with another student who has a phone if you don't have one that will work.

You will also need a laptop computer for development, with administrator access for installing tools. Windows and MacOS are both fine. Probably Linux is fine, too, but I haven't looked into that. You will be expected to bring your laptop to class. If you do not have a working laptop (either now or in the future), please contact ITS to work out a solution (e.g., a loaner laptop or some other arrangement).

Class Meetings

Time and Place: Fridays 8:30 AM to 11:30 AM in 1152 SPH2

Duration of Course: Every Friday from Sept. 6 - Dec. 6 (inclusive), with the exception of Nov. 22 (Thanksgiving).

End of Course: Thursday, December 6 is the final class meeting, and your physical presence will not be required for this course after that date. There is no final exam.

Class meetings will combine a lecture with a lab featuring hands-on coding. Generally speaking, most class meeting will be about half-and-half, with the lecture occuring first followed by the lab exercises. The particular structure of class meetings may vary somewhat from week to week.

You are expected to complete each week's readings before lecture. Lectures will often contain material and examples not present in the readings. Lab exercises can be completed outside of the class meeting, and in many cases some work will be required at home to complete them. Mere attendance at class meetings is not recorded or graded, though of course attendance is highly recommended.

Course Topics

Here is a provisional list of the topics we will cover in 669 this semester. The precise timing and sequence of these topics (and the topics themselves!) may change. Roughly speaking, each topic maps onto a week of the semester. Please refer to Canvas for the schedule of topics and assignments.

1. Intro
2. JavaScript and TypeScript
3. Angular
4. Ionic UIs / Storage
5. Mobile Design Principles and Standards
6. Testing and Debugging
7. Native Integration
8. The App Lifecycle and Notifications
9. Backends: Firebase

10. APIs, Maps, and Location
11. Analytics
12. Deploying to App Stores / Emerging Platforms

Communication

Feedback

We would love your feedback at any time on how the course is going for you. Comments can be submitted anonymously, or you can add your name and email address if you'd like a response. The Feedback form will be available all semester, and we will monitor responses on a regular basis.

Piazza

For questions about material in the course, including programming questions and general confusions about assignments, you should post on the **Piazza site**, which you can access from our Canvas site. Before posting, however, search and/or browse through previous answers to see if your problem has already been addressed.

Anyone may answer/respond to questions on the Piazza site, which will be monitored by course instructors. Specific answers to lab/project problems may *not* be posted on the Piazza site. Instructors will monitor this site to answer questions that your fellow students cannot/don't answer, to encourage continued discussion, and to provide additional explanations. We also encourage reading through the Piazza site semi-frequently; others questions and answers may be extremely helpful for you!

Do NOT post more than ~4 lines of code on Piazza, and do NOT post code that would reveal the answer to a lab or project component. If you're stuck on a coding problem, do your best to isolate the problem—this will not only keep you from sharing solutions, but it will also increase the likelihood that you will get a useful, relevant answer.

Office Hours

Course instructors will hold regular individual office hours. For Prof. Newman's office hours, you can book an appointment or just drop by—unbooked appointment slots are available on a first-come, first-served basis. If you book a slot but need to cancel, please release the slot on the calendar so that others will know it's available.

If you wish to meet with a specific instructor but can't make their designated office hours, contact the instructor directly by email to try to find another time. Be forewarned that your instructors are very busy, though, and it would be much, much better if you could make it during office hours!

In general, you should try to resolve your programming issues using searches and Piazza before taking them to office hours, in order to build your own problem solving skill and to help make efficient use of class resources.

At the same time, office hour conversations don't have to be about class assignments or material. Drop by, introduce yourself, and feel free to talk about general interests, personal projects, career plans, other courses, etc.

Grading Questions and Concerns

For questions or concerns about grades/scores ("when will my Project 4 be graded?" "I think I got an incorrect score on problem 3", etc), please **submit your concern** via **this google form**, also linked on the Canvas site. We will address those questions by responding to you via email, as soon as possible, but perhaps not immediately.

Email

For questions not related to course material, e.g. questions about extensions or personal circumstances, but NOT about regrades, please email the instructors' email list: si669-fall18-instructors@umich.edu. That will reach all members of the course instructional team (Prof. Newman, GSI, IA). This will increase the likelihood of receiving a timely response, and also will allow the instructional team to stay aware of how we are implementing course policies and communication.

Please do not ask an individual instructor about changing your grade(s). We make grading decisions as a team to ensure that everyone's grades are addressed in the same way.

Contacting Prof. Newman Directly

If you are uncomfortable sending an email to the entire instructional team for personal reasons, contact Prof. Newman directly via email: **mwnewman@umich.edu**.

If you email any instructor(s)/the instructor email list, we *can not* guarantee that you will get an email back within 48 hours, but we will try our best to do so. If you have not received a response after 3 days or so, it's reasonable to send a follow up to make sure your request did not slip through the cracks.

Canvas

General announcements about course scheduling changes, clarifications, important upcoming events, etc., will occur via Canvas Announcements. You should make sure your Canvas settings are set up so that you *will* receive email from Canvas Announcements for this course. **You will be expected to have received information sent via Canvas Announcements.** (The default Canvas settings will have you getting that email, but if you turned Canvas emails off, make sure you turn 'em back on or ask someone to figure out how to do so.)

All assignments, including expected readings, lab exercises, and projects will be accessible via the Canvas site. All grades will be available on the Canvas site. We will try our best to grade assignments as quickly as possible.

Assignments and Grading

There are four components of the course. Your performance on those four components will determine your grade.

Course Component	**Percentage of Final Grade**
Knowledge Checks	20%
Lab Exercises	20%
Projects	30%
Final Project	30%

Knowledge Checks

Knowledge Checks are brief online quizzes that are designed to make sure that you understand the key points that were covered in a particular course lesson. In SI 669, Knowledge Checks will be assigned on a weekly basis (with a couple of exceptions). Knowledge Checks are available online for you take at your convenience (before the due date, of course). You have up to three attempts to maximize your score on each Knowledge Check--your highest score will be kept. You can take as long as you like for each Knowledge Check, and you are free to review the relevant readings and lecture notes, along with any other materials you find useful for answering questions.

Your lowest Knowledge Check grade will be dropped. The remaining Knowledge Check grades will be averaged to compute the Knowledge Check component of your course grade.

Lab Exercises

There will be a lab exercise assigned each week to give you hands-on experience with that week's concepts. Lab exercises are designed to be started during class meetings, where help is available from the GSI and instructor. Lab exercises are designed to be completed with 2-3 hours of work, which means that they will often not be completed during class meetings, and will require an hour or two of work outside of class. Lab Exercises will be due pretty much every week, regardless of whether there is also a project due soon, so please plan accordingly.

Some Lab Exercises may offer the opportunity to earn extra credit. Your lowest Lab Exercise grade will be dropped. The remaining Lab Exercise grades will be averaged to compute the Lab Exercise component of your course grade.

Mini-Projects

There will be 3 mini-projects, each designed to give additional experience with key course concepts. You will have approximately two weeks to work on each mini-project, and the projects will build upon examples that were covered in readings, lectures, and lab exercises.

Some Mini-Projects may offer the opportunity to earn extra credit. All Mini-Projects count equally, and none will be dropped. The Mini-Project grades will be averaged to compute the Mini-Project component of your course grade.

Final Project

There will be a final project in this course that you get significant time to work on and build up to, using a number of the concepts we'll address in the course. There will be structure and requirements, but you can also use creativity to pick what you will examine for your final project. You will give a demo of your final project during the final class meeting on Dec. 6. There will be no extra credit offered for the Final Project.

The Final Project has several milestones, each of which contributes to your Final Project grade. These components are weighted as follows:

Project Component	**Percentage of Project Grade**
Final Project Proposal	10%
Final Project Product Backlog and Release 1 Plan	10%

Final Project Checkpoint: Release 1 results and Release 2 planning	10%
Final Project Demo	35%
Final Project Code and Report	35%

Letter Grades

Your final grade in the course will be based on the weighted percentage of your grades across each component. Due to extra credit, it's possible to get more than 100%.

A+	>99 %
A	94-98.99%
A-	90-93.99%
B+	87-89.99%
B	83-86.99%
B-	80-82.99%
C+	77-79.99%
C	73-76.99%
C-	70-72.99%
D+	67-69.99%
D	63-66.99%
D-	60-62.99%
E/F	0-59.99%

The cutoffs between letter grades are hard cutoffs. Even if you missed a cutoff by a few tenths of a percentage point, please do not ask to be bumped up a level. If you feel you have been graded unfairly on a particular assignment, you may submit a regrade request, following the process described in "Communication."

Late Policy, Extensions, and Excused Absences

Any assignment may be submitted up to 4 days late, at a deduction of 10% of your grade per day of late submission.

Things come up. We understand that not everyone will be able to get everything done on time and there may be reasons why an assignment needs to be turned in a bit late. Here are the policies regarding late assignments:

- Late assignments will be assessed a 10% penalty for each day that they are late. Assignments that are more than four days late will not be accepted.
    - Note that the penalty is calculated based on the number of points available, not on the number of points awarded. So, for example, if an assignment that earned 88 out of 100 available points is turned in 46 hours late, the points awarded would be 88-20=68.
- To spell it out:
    - Before the deadline: No penalty
    - 0:01-24:00 hours after the deadline: 10% off
    - 24:01-48:00 hours after the deadline: 20% off
    - 48:01-72:00 hours after the deadline: 30% off
    - 72:01-96:00 hours after the deadline: 40% off
    - > 96 hours after the deadline: 100% off

We reserve the right to change this with advance notice (e.g. if we will use the solution to an assignment in an upcoming lecture that is less than 4 days away). We will let you know about this with as much notice as possible.

Canvas will show you when your assignment is late -- it's any time after the deadline! Yes, even ten minutes. Gotta draw the line somewhere. It's good training for absolute software deadlines, should those be things you encounter in your future. (We will calculate how many days late it is. You can do that, too, of course.) After four days, we will no longer accept it.

Extensions and Excused Absences

In order to maintain as much fairness as possible in this course, except in extraordinary circumstances, we do not generally give extensions or excuse assignments in this course. Extensions and excused assignments may be granted for *extreme personal circumstances* only. ”Extreme personal circumstances” include illness, family emergencies, and significant emotional distress (in this case, please also see “Mental Health and Well-Being at the University of Michigan” below). The legitimacy of each specific instance is at the discretion of the teaching team. "Legitimate reasons" do not include awesome parties, family trips, or excessive course-related or extracurricular work—you are required to plan ahead to meet your deadlines in all classes. The secret word is swordfish.

If extreme personal circumstances occur and you believe you need an extension, make a request to the Teaching Team via email for the extension **as early as possible** and you will receive a response as soon as we can get back to you.

The ability to drop a few homeworks, discussion section attendances, and lecture exercises are

there for things like having a job interview or extracurricular event that overlaps with a class meeting, feeling tired, going on vacation early, starting last-minute, etc. Please don't ask for excused absences for those. **Extensions and excused absences will *not* be granted for any event that you could plan for in advance.** Any major personal event that has advance planning (birthdays, weddings, graduations, for example) or vacations that do not align with the university's holiday schedule (available here) are things you should plan for *in advance*. Extensions will *not* be granted for job interviews or conference travel, even though we're happy you get to experience those.

Look ahead at the course schedule now. If you see a real conflict, you should tell your instructors as soon as possible (and no later than the 3rd week of class) so you can make a plan ahead of time. But if, for example, you know you will be celebrating a birthday one weekend, you should just plan to get started on your work ahead of time that week; we will not offer an extension, essentially because we cannot reasonably do that for everyone.

Religious holidays you observe that do not accord with the university's holiday schedule ARE, per university policy, always excused absences. If a holiday will take you away from class (a class meeting), please speak with the instructional team in advance, to find a time to make up for any content you miss. We will work with you to arrange an alternative.

For other accommodations you need to take this course, please see the section on **Accommodations and Services for Students**.

Academic Integrity & Collaboration

This course includes both individual and group assignments. In the case of individual work, you may not include code written by others. In the case of group work, you may not include code written by others who are not in the group. You may, however, discuss general solutions to programming problems with classmates and colleagues--these discussions should not include sharing or demonstrating specific code solutions. You may also search online for examples that are relevant to the problem you are trying to solve, but you may not copy and paste code directly into your source files (retyping a few lines from online examples is OK--retyping helps you to slow down and make sure you understand what the code is doing. If you don't understand what it's doing--get help!). Such examples should be cited in your code (see below) to give credit to the original authors, and should not constitute a significant percentage (i.e., more than about 25%) of your total solution.

The restrictions on that are as follows:

- If you get help from someone outside your group for writing your code, cite that specifically in your submission (in a comment in your program). You do not need to cite learning from lecture or section, or from your textbook(s) or instructors (unless you are citing collaboration with another student that occurs during lecture or section!). If you use

another person's code directly, in class or from the internet, you must also cite that in a comment to your code indicating what you borrowed, and where/who from. Note that the definition of who is “in your group” is relative to a particular assignment--if it is an individual assignment then nobody is “in your group.”

- If you give help to someone else, or work with others on HW, etc, *do not type on their computer*. Talk as much as you like, but everyone should get the experience of typing and completing problems in the way they normally do (e.g. it's not okay for a friend in the class to type your HW for you while you watch because they know how to do it already, nor for you to ask for someone's computer to just finish a problem because it's frustrating not knowing how to communicate what you're thinking about code. The challenge is worth it!).
- Posting small code snippets (generally 4 lines or less) on Piazza is great, and encouraged! Posting complete answers to HW problems is not acceptable.
- Study groups are welcomed and strongly encouraged, as is talking through any problems you encounter. However, if you feel you are either giving or receiving all of/too many of the answers, it is probably time to break up the study group. Please contact Prof. Newman and/or your GSI confidentially if you have a problem like this and are having difficulty dealing with it on your own.
- Using answers provided directly from past semesters, if they are the same or similar to work this semester, is not acceptable. To use any past assignments from others as reference for your own is a serious breach of academic honesty and may result in serious consequences. Copying others' answers is very different from *hearing an explanation* from a fellow student or a past student, even if the explanation involves walking through little bits of code.

Any statements of phrases from the work of others, including code snippets, must be clearly identified as quotations, and proper citation must be provided. Unless otherwise specified, all submitted work must be your own, original work.

The format for citing code is as follows:

<author(s) names> (<date>) <title of program/source code> (<code version>) [<type>]. Web address or publisher.

E.g.

// Smith, J (2011) GraphicsDrawer source code (Version 2.0) [Source code].
// http://www.graphicsdrawer.com

However, for citing fellow classmates' code, it's OK to say in your assignment, in a code comment, e.g.

```
// I worked on this code lines 10-15 with Jane Doe.
```

In such a case, Jane should also comment her assignment about you! (You do not need to cite working with a GSI or IA or Prof. Newman.)

Any violation of the School's policy on Academic and Professional Integrity (stated in the Master's and Doctoral Student Handbooks) will result in serious penalties, which might range from failing an assignment, to failing a course, to being expelled from the program. Violations of academic and professional integrity will be reported to UMSI Student Affairs. Consequences impacting assignment or course grades are determined by the faculty instructor; additional sanctions may be imposed by a school administrator.

Accommodations and Services for Students

If you need or believe you may need an accommodation, e.g. for a disability, please let the instructors know at your earliest convenience. Some aspects of this course, the assignments, the in-class activities, and the way we teach may be modified to facilitate your participation and progress. As soon as you make us aware of your needs, we can work with the Office of Services for Students with Disabilities (SSD Office) to help us determine appropriate accommodations. SSD (734-763-3000; http://ssd.umich.edu/) typically recommends accommodations through a Verified Individualized Services and Accommodations (VISA) form. We will treat any information that you provide in as confidential a manner as possible.

Please make sure Prof. Newman receives all notification about accommodations (e.g. test accommodations).

If you have a concern along with or separate from the above that affects the process of class at large or for you, please approach or contact Prof. Newman confidentially. I (Prof. Newman) will also treat any such shared information in as confidential a manner as possible and will do what I can to ensure you have what you need in this course and/or have the resources to find it.

Mental Health and Well-Being at the University of Michigan

The University of Michigan is committed to advancing the mental health and well-being of its students, while acknowledging that a variety of issues, such as strained relationships, increased anxiety, alcohol/drug problems, and depression, directly impacts students' academic performance. We take this seriously.

If you or someone you know is feeling overwhelmed, depressed, and/or in need of support, services are available. For help, contact Counseling and Psychological Services (CAPS) at (734) 764-8312 and https://caps.umich.edu/ during and after hours, on weekends and holidays or through its counselors physically located in schools on both North and Central Campus. You

may also consult University Health Service (UHS) at (732) 764-8320 and https://www.uhs.umich.edu/mentalhealthsvcs, or for alcohol or drug concerns, see www.uhs.umich.edu/aodresources. For a more comprehensive listing of the broad range of mental health services available on campus, please visit: http://umich.edu/~mhealth/

If you are seeking advice, answers to questions, or help accessing resources, you should contact the Office of Academic and Student Affairs within UMSI, on the 5th floor of the Collegian building, 333 Maynard St. You can also contact them for support with academic or personal advising while in the program.