

CMSI 2021

WEB APPLICATION DEVELOPMENT

<http://dondi.lmu.build/fall2026/cmsi2021>

Fall 2026—Pereira 121; 2 semester hours
MW 10:50am–12pm (01); MW 12:15–1:25pm (02)

Office Hours MW 5–6pm, TR 3–5pm,
or by appointment (*don't hesitate to ask!*)

John David N. Dionisio, Ph.D.
dondi@lmu.edu

Doolan 102
(310) 338-5782

Objectives and Outcomes

This is a two-unit practicum course, meaning you will receive training and gain experience building systems according to modern best practices. All computer science majors must complete this or Mobile Application Development (CMSI 2022) in order to experience full-stack development *before* undertaking the rigorous courses in the junior and senior years. Long after the course concludes, my hope is that you will be able to:

1. Complete web application projects as full-stack developers
2. Demonstrate useful skills in front-end, back-end, API, and database development
3. Participate in an agile development process
4. Demonstrate competency in a web application stack, using modern frameworks such as React
5. Show knowledge of processes for envisioning and communicating your vision for an application or product, such as ideation, pitching, and presenting

In addition to the course-specific content, you are also expected to:

6. Follow disciplinary best practices throughout the course

Prerequisites/Prior Background

You should have knowledge and experience equivalent to at least one semester of computer programming, but more importantly you must bring: (1) a growth mindset, (2) a respectful attitude, and (3) a willingness to work hard, to never give up, to accept feedback, and to strive to avoid imposter syndrome and stereotype threat. Make sure you take the time for effective self-study. Interact fully with classmates and teammates: the computing industry is one of the most collaborative fields in which to work, and your course experience should reflect this.

Materials and Texts

Most of the course content will be found online—and no different from what actual web app developers use for their own learning. It's a practicum course after all! Links to these resources are on the course website. In addition, do not hesitate to look for further information regarding the concepts, techniques, tools, and paradigms in the course.

As a practicum, we focus on specific technologies to keep things cohesive and concrete. For this offering, these are the React on the front end and a hosted database on the back. Code will be written in CSS, JavaScript, and JSX (which looks a lot like, but isn't exactly, HTML). As such, you'll need:

- A web browser (pretty sure you have one of those already 😊)
- A source code editor, ideally one that supports the Prettier code formatter plug-in
- A GitHub account for storing your code
- Credentials for use with a hosted database service (specifics to be determined later)

Course Work and Grading

Your final grade will be based on the percentage of the points you get for the following milestones and your demonstrated knowledge & understanding of those milestones, with these total points:

Standalone single-page web app	200
Generic API-backed web app	200
Database-backed web app	200
Your own web app	400

Total 1000 points

Percentages $\geq 90\%$ get an A– or better; $\geq 80\%$ get a B– or better; $\geq 70\%$ get a C– or better. I may nudge grades upward based on qualitative considerations such as degree of difficulty, effort, class participation, time constraints, and overall attitude toward the course.

Inherent Knowledge & Understanding

The degree that you will eventually attain is meant to reflect your *knowledge & understanding* of the discipline represented by that degree. The work artifacts you generate along the way are *means* to this end, and not the ends in themselves. This is the key difference between a class and a job: on the job, you are expected to produce but in class, you are expected to learn. And learning comes first: your future ability to produce is bounded by what you have learned and what you will be able to learn.

As such, although producing the course's deliverables contribute to your grade, these deliverables alone comprise half of the points for each milestone. The other half represents how well you have learned the underlying principles that lead to such deliverables—i.e., your *inherent knowledge & understanding*. Engagement in class and office hours—the questions you ask, the answers you provide, the observations you make—will contribute points to this category over time. As such, this engagement is essential and meaningful, and you will genuinely miss out if we miss you.

To this end, class discussions will work to ensure that everyone has a chance to express themselves in this manner. If you prefer a smaller, more private setting, make sure to drop by office hours so that you can have this opportunity.

Demonstrating your inherent knowledge & understanding includes, but is not limited to:

- Assimilating and using the core concepts and vocabulary of a subject area
- Identifying the boundaries of your current understanding and forming questions to help expand those boundaries
- Formulating novel solutions without a starter template or pattern, and validating their results
- Expressing, questioning, and navigating decisions or tradeoffs
- Reading, understanding, and providing feedback on work products created by you or by others

All this must be doable *without crucial reliance on technology or tools*. The forward-looking thread that connects all of these activities is that they align with how you would demonstrate and apply this knowledge beyond the classroom. Even if not every class you take will carry over to your life trajectory, *during* the course you are expected to approach the subject in that manner.

Disciplinary Best Practices (Objective 6)

Appearance of these in the syllabus constitutes fair warning of the consequences of not heeding them.

Clean, Understandable, & Understood Code

The *quality* of your code and your ability to communicate about it plays a part in determining your grade—it isn't solely about its *correctness*. You are responsible for everything you commit. With group work, you are responsible for everything that your group commits.

Version Control

Speaking of commits: *version control* is an indispensable part of today's computer science landscape in industry, the academe, and the open source community. Make sure that you get the hang of it.

None of the assignments can be completed (well) overnight; they should be the result of steady progress from the moment they are assigned to the date they are due. "One and done" submissions will negatively affect the final score.

Punctuality

An assignment's number is its due date in *mmdd* format, due by 11:59:59.999pm of that date. If you primarily work in the Keck Lab, plan on finishing at home or a few hours earlier: the lab isn't officially open that late so access is not guaranteed.

Point values are based on the state of your assignments at that moment. If there is an issue that prevents you from submitting on time (e.g., personal or family issues, sickness, conferences, can't-miss raids, job interviews, family travel, or emergencies), let the instructor know ahead of time.

Workload Expectations

In line with the *LMU Credit Hour Policy*, the workload expectation for this two-credit-hour course is a minimum of $2 \times 3 = 6$ hours of work by an average student per week, including the time that we spend together in the classroom.

Attendance

Attendance at all sessions is expected for inherent knowledge & understanding credit. If you must miss class, it is your responsibility to notify me about this and keep up with the course.

The last day to add or drop a class without a grade of W is September 4. The withdrawal or credit/no-credit deadline is November 13.

Academic Integrity

Loyola Marymount University is a community dedicated to academic excellence, student-centered education, and the Jesuit and Marymount traditions. As such, the University expects all members of its community to act with honesty and integrity at all times, especially in their academic work. Academic honesty requires that all members of the LMU community act with integrity, respect their own intellectual and creative work as well as that of others, acknowledge sources consistently and completely, act honestly during exams and on assignments, and report results accurately. As an LMU Lion, by the Lion's Code, you are pledged to join the discourse of the academy with honesty of voice and integrity of scholarship.

Academic dishonesty will be treated as an extremely serious matter, with serious consequences that can range from receiving no credit for assignments/tests to expulsion. It is never permissible to turn in any work that has been copied from another student or copied from any source (including the Internet) without properly acknowledging/citing the source. It is never permissible to work on an assignment, exam, quiz, or any project with another person unless your instructor has indicated so in the written instructions/guidelines. It is your responsibility to make sure that your work meets the standard of academic honesty set forth in the "Academic Honesty Policy" found at:

<https://academics.lmu.edu/honesty>

Technology Use and Academic Integrity

The use of technology in alignment with academic integrity serves the goal of attaining inherent knowledge and understanding:

- ✓ Technology may be used for *instrumental* help: building skills, learning independently, etc.
- ✗ Technology may *not* be used for *executive* help: generating work without understanding, short-circuiting your agency & accountability, etc.

Examples of executive help include:

- ✗ Generating work artifacts that you do not understand yourself
- ✗ Fixing defects without personally knowing their root causes and the nature of the fix
- ✗ Copy-pasting content from other sources without being able to explain what it does, how it works, nor why it is the best alternative

Examples of instrumental help include:

- ✓ Seeking explanations to gain understanding
- ✓ Navigating work products to think about them critically and produce them on your own
- ✓ Accelerating rote, routine, or mechanical tasks that you could otherwise do yourself

TL;DR *Never use technology (including but not limited to AI) to avoid learning, but do use it to augment, amplify, or accelerate (your) learning.*

Course Evaluations

Student feedback provides valuable information for continued improvement. All students are expected to fairly and thoughtfully complete a course evaluation for this course. This semester, course evaluations will be administered online through the Blue™ evaluation system. You will receive an email notification at your Lion email address when the evaluation form is available. You may also access the evaluation form on Brightspace (<https://brightspace.lmu.edu>) dashboard during the evaluation period. Your responses will be anonymous and will not be linked to you in any way.

Special Accommodations

The Disability Support Services (DSS) Office offers resources to enable students with ADD/ADHD; physical, learning, and psychiatric disabilities; and those on the autism spectrum to achieve maximum independence while pursuing their educational goals. Staff specialists interact with all areas of the University to eliminate physical and attitudinal barriers. Students must provide documentation for their disability from an appropriate licensed professional. Services are offered to students who have established disabilities under state and federal laws. DSS personnel also advise students, faculty, and staff regarding disability issues. Students who need reasonable modifications, special assistance, academic accommodations or housing accommodations should direct their request to the DSS Office as soon as possible. All discussions will remain confidential. The DSS Office is located on the 2nd floor of Daum Hall and may be reached by email at dsslmu@lmu.edu or phone at (310) 338-4216. Please visit <http://www.lmu.edu/dss> for additional information.

Topics and Important Dates

Specifics may change as the course progresses. University dates (italicized) are less likely to change.

September	HTML/CSS/JavaScript; React; network, authentication, and database APIs
<i>September 4</i>	<i>Last day to add or drop a class without a grade of W</i>
<i>September 7</i>	<i>Labor Day; no class</i>
October	Data storage and persistence; Material UI; React Router
November	Testing, context, server/cloud deployment, additional topics as time allows
<i>November 13</i>	<i>Last day to withdraw from classes or apply for Credit/No Credit grading</i>
<i>November 25–27</i>	<i>Thanksgiving break; no class</i>
December	Ideation and design; pitch presentations and work sessions
<i>December 16, 18</i>	<i>Final web app presentations (based on LMU final exam schedule)</i>
<i>December 18</i>	<i>Final web app milestone due</i>

You can view my class calendar and office hour schedule in any iCalendar-savvy client. Its subscription link can be found on the course web site (it's too long to provide in writing).

Tentative Nature of the Syllabus

If necessary, this syllabus and its contents are subject to revision. Students are responsible for any changes or modifications announced or distributed in class, emailed to students' LMU Lion accounts, or posted on LMU's course management system, Brightspace. If you are absent from a synchronous class session, it is the student's responsibility to check Brightspace and with the professor to see if you missed any important class announcements. Students should not rely on word-of-mouth from classmates.