

CS 343 Application Development

James Madison University, Spring 2025

Have you ever wanted to create your own mobile app, interactive web application, or video game? This hands-on course will build the foundation for learning these and other skills. All of these are “applications” (rather than e.g. “services” or more generic “systems” or “processes”). The technologies that facilitate application development change rapidly over time, and which are best for a given application can vary.

This course will focus on fundamental concepts and skills that are common across different platforms. The course will also introduce a few specific technologies that are currently important and popular, and likely to remain so. The course will also introduce the git version control system, which is widely used in industry and academia.

Course & Instructor Information

Website:	https://w3.cs.jmu.edu/kirkpams/343/	
Time/Place:	M/W/F 1:50 – 2:40 PM	
Textbooks:	<i>Fundamentals of Web Design, 3rd Edition</i> by Connolly and Hoar	
Instructor:	Prof. Michael S. Kirkpatrick	Email: kirkpams@jmu.edu
Office:	King 223	Phone: (540) 568-3371
Office Hours:	M 2:45 – 4:30 PM, Th 2:00 – 4:00 PM, F 11:15 AM – 12:30 PM	

Course Structure and Grading Policies

Each week begins with a pre-class reading assignment and accompanying prep activity. Classes will start with a warm-up question, then proceed to a mix of mini-lectures, pair programming exercises, and multiple-choice clicker questions (ConcepTests). Friday classes will include time to discuss the 12 programming labs. There will also be a semester-long project, two midterm exams, and a final. The following policies describe each component in more detail:

- **Weekly preps** - Prep activities are due before class on Monday (12:00 PM). Before that time, you can attempt these activities as many times as you need. Your lowest two preps will automatically be dropped.
- **Participation** - Participation points will be earned through warm-up questions and ConcepTests in class, and Canvas metacognitive reflections, which are due each week by Thursday at 3:00 PM; late submissions will not be accepted.
- **Programming labs** - Each Friday will include time to discuss and/or work on the lab, which is due by the following Monday at 11:59 PM. Each lab will be graded based on completion (2 points). Submissions that pass some but not all of the tests will be given 1 point.
- **Project** - Throughout the semester, you will work on developing an application of your own. (If you are completely stuck, we will provide possible options to consider.)

During the semester, you will be required to meet certain milestones, including a preliminary design, a prototype, and a final demo. Details will be provided via the course web site.

- **Midterm and final exams** - All exams will take place during the designated class time and they will contain both conceptual and code writing problems. The final will be worth 50% more than a midterm exam.

This course uses a specifications-based grading approach that may differ from a typical numeric scale that you may be used to. Each grading component is evaluated only as a letter grade with no plus/minus adjustments and no rounding. (E.g., an average of 84.7% on exams will be evaluated as a B.) Course letter grades are based on earning that letter grade in all four components. The following table describes each component's grade level requirements.

A requirements	
Exams:	Average of 85% on exams
Project:	Requirements are described as part of the project specification
Labs:	Earn 20 lab points
Others:	Grade of 90% or higher on participation and preps
B requirements	
Exams:	Average of 75% on exams
Project:	Requirements are described as part of the project specification
Labs:	Earn 18 lab points
Others:	Grade of 80% or higher on participation and preps
C requirements	
Exams:	Average of 65% or higher on exams
Project:	Requirements are described as part of the project specification
Labs:	Earn 15 points, with 2 points on at least 6 labs
Others:	Grade of 70% or higher on participation and preps

Many students fall between these requirements, such as satisfying the A requirements for labs and projects but getting a B on the exams. Split grades will be determined as follows:

- If all are a mix of As and Bs:
 - If the exam and project grades are both As, the course grade will be A-.
 - If two of the four are an A and that includes either the exams or project, the course grade will be B+.
 - Otherwise the course grade will be B.

- If all are a mix of As, Bs, and Cs:
 - If three of the four are As or Bs, or if both the exams and project grades are As or Bs, the course grade will be B-.
 - If two of the four are a As or Bs and that includes either the exams or the project, the course grade will be C+.
 - Otherwise the course grade will be C.
- If any component falls below the C requirements, course grades will use a traditional 4.0 scale with 35% for the exams, 25% for the project, and 20% each for the labs and others.

The following scenarios illustrate possible grades:

	Student 1:	Student 2:	Student 3:	Student 4:	Student 5:
Exams	A [91%]	A [86%]	B [78%]	C [68%]	C [70%]
Project:	A	B	C	B	C
Labs:	B [19]	B [18]	B [18]	C [15]	F [0]
Others:	A [95%]	A [90%]	A [90%]	B [85%]	C [75%]
Course grade:	A-	B+	B-	C+	D+

Adjustments or extensions will be granted based on extraordinary circumstances at the instructor's discretion. If you are sick, please do not come to class; I assure you that the missed participation points will not impact your grade.

Course & University Policies

- **Classroom inclusion** - Learning environments should be built on mutual respect and support a diversity of thoughts, perspectives, experiences, and identities. Please advise me regarding personal circumstances (including your name's proper pronunciation, any name or gender pronouns not reflected on MyMadison, or significant extracurricular commitments) that may be relevant to your full participation in class.
As computing professionals, we adhere to the ACM Code of Ethics and Professional Conduct (<https://www.acm.org/code-of-ethics>), which forbids discrimination and harassment of all types. If you feel someone is violating these principles (including inappropriate or demeaning jokes), it is your responsibility to take action by informing me or (if you feel comfortable doing so) addressing the individual directly. I will do my best to preserve your confidentiality while addressing the issue.
- **Laptop policy** - This course is structured to use class time for discussions and other in-class activities. You may use a laptop for on-task use only. **Do not work on projects or other assignments, for this class or others, during class time.** If your laptop use becomes a distraction to your peers or to me, you will no longer be able to use it in class.

- **Attendance and grading** - Attendance is not recorded directly, but is necessary for participation credit. Please let me know if you have any specific obligations (such as athletic team commitments or military service) that may constrain your attendance.
- **Communication policy** - Communication outside of class will be primarily through office hours and Piazza (available through Canvas). My email should only be used for documenting things, such as absences or regrade requests. (I will direct all non-private course-related questions to Piazza.) I have designated times each day for responding to course communications and I am generally unavailable outside these times.
- **Academic integrity** - Students are expected to comply with the JMU Honor Code as stated in the Student Handbook and available from the Honor Council Web site at <http://www.jmu.edu/honor/code.shtml>. The Honor Code states it is a violation to “render *unauthorized* assistance to another student by knowingly permitting him or her to see or copy all or a portion of an examination or any work to be submitted for academic credit.” In the context of this course, this portion of the Honor Code means:
 - You are allowed to collaborate on labs, provided that you contributed to any code that you submit; all such collaborations must be clearly identified in comments on submitted code. You are allowed to work **in pairs** on projects; no collaboration outside of pairs is allowed. Except for these two exceptions, **copying and/or sharing code is expressly forbidden**. This includes copying code from previous semesters’ solutions (including your own or other students’) or other external sources. It also includes posting your code publicly, such as on Github.
 - It is acceptable to consult other resources (e.g., Stack Overflow) for clarifying examples but not wholesale copying of significant pieces of code. All such references must be documented explicitly within code comments. However, the wholesale copying and pasting of the project descriptions or source code into generative AI tools (e.g., Copilot, ChatGPT) is strictly prohibited.
 - Extensive discussions with other students that are likely to lead to similar code must be disclosed *before or during submission* (either in person or documented in code comments). Unintentional violations will be granted leniency, though a penalty may still apply.
- **Adding/dropping classes** - You are responsible for registering for classes and for verifying your schedule on MyMadison. Deadlines for adding or dropping classes are available from the JMU Registrar.
- **Cancellations** - JMU’s cancellation policy (<http://www.jmu.edu/JMUpolicy/1309.shtml>) provides details regarding inclement weather and other emergencies.
- **Religious observance accommodations** - All faculty are required to give reasonable and appropriate accommodations to students requesting them on grounds of religious observation. If you need to request accommodations, you must let me know at least 2 weeks in advance.

- **Disability accommodations** - JMU abides by Section 504 of the Rehabilitation Act of 1973 and the Americans with Disabilities Act, which mandate reasonable accommodations be provided for students with documented disabilities. If you have a disability and may require some type of instructional and/or examination accommodations, please contact me early in the semester so that I can provide or facilitate provision of accommodations you may need. If you have not already done so, you will need to register with the Office of Disability Services, the designated office on campus to provide services for students with disabilities. The office is located in Wilson Hall, Room 107 and you may call 540-568-6705 for more information.

Course Catalog Description

Overview of application software development fundamentals and their use in building stand-alone applications, visualizing and interacting with complex data representations, and controlling the Internet of Things. This course provides an introduction to the technologies used to create modern user interfaces (e.g., web technologies), establishing a common foundation for later application-focused courses. *Prerequisites: Fully admitted Computer Science majors or minors only and a grade of "C-" or better in CS 159.*

Detailed Course Objectives

Following the successful completion of this course, students will be able to:

- Summarize the key steps for publishing, retrieving, and displaying web documents.
- Justify and apply accessibility standards for web documents.
- Apply accessibility standards to web documents.
- Generate and manipulate DOM objects.
- Describe the benefits of functional programming in data manipulation.
- Evaluate the advantages and disadvantages of client-side storage technologies.
- Dynamically generate and manipulate images based on structured data.
- Create dynamic asynchronous event handler software that avoids common timing-related errors.
- Describe how authentication, authorization, and access control can mitigate common web application risks and threats.
- Apply common web-based techniques to mitigate common risks and threats arising from untrusted input and third-party software.
- Identify and consult credible software documentation and tutorials for learning new technologies.