# Technological Feasibility Analysis


## DIGital

**Sponsor:**
Dr. Kayeleigh Sharp

**Faculty Mentor:**
Scott LaRocca

**Team Members:**
Devin Jay San Nicolas
Tate Whittaker
Ryan Wood
Jarom Craghead

October 24, 2025

- ☐ **Cover Page and TOC:** you should have the customary cover page for your report. If any project document is longer than five pages (which this one most certainly is), then you should include a Table of Contents as the second page, outlining the major sections (but not the subsections).

- ☐ **Introduction:** Every single document related to a project should have an introduction! Think of our storytelling metaphor! In this introduction, you are briefly going to pitch your project: remind people why it's important. Here is the flow:
  - ☐ **Context:** Start with the "big picture" paragraph. Why should the world care? Stuff like "Over 50 million Americans suffer from diabetes and it costs our healthcare system XX billion/year" for a project related to diabetes management, or "Species extinction is wiping out xx species a year" for a project related to monitoring/addressing natural preservation. Often, you can get a start on this from your mini-intro knowledge. After finishing this section, your reader should be thinking, "This is a really serious and important issue!"
  - ☐ **Problem:** Then talk about how the problem you invoked is currently being dealt with. Start with how current solutions look in general, and then introduce your sponsor and how the role their outfit plays in the process, i.e., how they specifically have been trying to solve the problem. Then narrow it done to the problems your sponsor has: how, if at all, has your sponsor been managing so far? What is their current solution? You'd want to highlight how inadequate/inefficient/labor-intensive the current approach is. Be specific: this is also your chance to educate the reader about how the business process of your sponsor works. After finishing this paragraph(s), your reader should be thinking "Wow, what a mess...something needs to be done!"
  - ☐ **Goal:** Introduce your solution vision. Start with a general sentence or two that describes your overall product/approach (i.e., we're building a web app/mobile app/whatever) and what it does in general (e.g., a value proposition). Then go on to highlight a few of the key features/functions; it should be immediately obvious to the reader that these proposed features are exactly what is needed to address the client problems you outlined earlier!
  - ☐ **Impact:** Finish the introduction by stating the expectations to improve the client's business. How does your solution address the issues presented before? How will the context be improved when your solution exists? The introduction must end with the feeling that a mission is ready to be accomplished.

You should see this project intro section as an investment for your entire capstone year: you can essentially re-use this intro section on all project documents, refining it a bit more each time, until it appears on your final project report! In terms of length, it's hard to do a good intro in less than 0.75 pages, and if it stretches beyond about 1.5 pages, it's too lengthy.

- ☐ **Segue into the main document body:** A "segue" is a transition, or bridge, between the sections of a document. Think of them as the glue that smooths the way between topics and helps the whole thing flow. You'll want them throughout your document, as you transition from talking about one thing to talking about another. They often sound something like this "Now that we have established that X, Y, we are ready to turn to a more careful analysis of Z. In the next section, we <outline topics>." As your first segue from the intro discussion into the main document, you'll want to turn from talking about the project in general to telling the reader what role the upcoming document plays and how it is structured. So something like "At this early stage in the project, we are in the process of analyzing the key technological challenges, identifying possible alternatives, and selecting which of those alternatives are the most promising solutions." Ok, good, now go on to briefly outline how your discussion will be organized: "In this Technological Feasibility Analysis document, we begin by analyzing the major technological challenges we expect <blah blah>; in the subsequent major subsections, we then analyze each of these areas carefully in turn, looking at alternatives, how we explored these, and rationale for choosing a particular solution; in Section Z, we then...". You get the picture: outline how your argument will flow.

Having outlined your story in this way, you now just launch into providing the flesh on the bones:

- ☐ **Technological Challenges:** In this section, you will outline all of the major technological needs/challenges that you see facing in your project. Of course, you may discover others later, but the whole point of this exercise is to think carefully at this early point, to avoid those surprises later on. So every major piece of the system that you can envision having to deal with, based on what you know so far, should be covered. These are essentially high-level requirements for your system, which raises the question of "how are we going to implement that?" The detailed contents here will, of course, vary widely depending on the project details; no two will be identical within our class! Just as examples, you might have things like:

- We will need a secure interaction and user authentication on our web app.
- We will need a way to communicate between our mobile app and the embedded hardware controller.
- We will need a way to provide a highly reactive interface in order to do <outline what specific functions require this> in our product.
- We will need a reliable data store backing our product that is capable of secure, highly accessible storage of <quantity> of data.

- We will need a way to keep the mobile apps synchronized to a level of fidelity of X seconds of delay between all of the users connected to the same group.
- We will need a way to spatially lay out a graphical genetic map based on a large database of genetic markers and their genomic positions.

You get the picture: every team will need to brainstorm to think about the specific technological hurdles they need to overcome. The emphasis here is on "major" and on "hurdles". There's no need to worry or talk about design details that are minor and unproblematic. You're not doing the full design here; you're trying to highlight and treat the major design decisions.

- ☐ **Technology Analysis:** This is the meat of your whole feasibility analysis; it constitutes the bulk of this document! In the intro/segue to this section, you will introduce the major technological issues or design decisions that you have identified as critical to project success in your early project explorations with the client. In the following major (3-5 page) subsections, you will then work through each of these issues in detail, describing the issue, your detailed analysis, and its outcome. Specifically, each subsection will likely have its own subsections as follows:
  - ☐ Introduce the issue (Intro paragraph).
  - ☐ Talk about the issue/challenge in a bit more detail, and give some detail about what functions you see needing to be provided and the role that you see them playing in the final product.

- ☐ **Desired Characteristics:** Now that we understand the design decision or tech challenge overall, talk about what an ideal solution would look like. In particular, you want to introduce the key characteristics that an ideal solution would have that are directly relevant to your particular project. This will vary by project, of course, but some typical characteristics might be licensing cost, speed/performance, ease of maintenance, existing user base, how mature the tech is, and so on. For each issue you identify, you must make it clear why this characteristic is important for your particular project. Just because someone else used this solution doesn't mean it's a good fit for your project! These characteristics are vital: they become the metrics on which you will be evaluating the alternative solutions.

- ☐ **Alternatives:** Now introduce the possible approaches to addressing this particular issue. For each one, have a little paragraph where you introduce the alternative, and say a little about how you found it (e.g., comparing approaches on forums, client recommended you look at it, whatever). Then give a few overview sentences with some general background on that product/package/framework: who developed it, how long it has been around, who has typically used it before, and in what kinds of applications. Not a major novella, just helps us understand a bit about this product.

- ☐ **Analysis:** Now you're ready for the important part: describe how you evaluated each of the alternatives on each of the criteria (desired characteristics) that you outlined earlier. Evaluation could take many forms: maybe you installed each of the alternative packages and subjected them to some basic testing, or maybe programmed a little 'hello world' trial in each of them; if your criterion was security, maybe you dug into the security features that each option provides and/or investigated how it ranks of various security forums. There is no magic formula; it will vary depending on the project and the criteria chosen. The key is that you show that you have done a reasonable and logical investigation of each alternative on each criterion, versus flipping a coin and choosing one!

- ☐ **Chosen approach:** To end the subsection, you bring your investigations on that technological choice to a decision. Start with a few summary sentences on what you found: pros and cons of the various options. Then bring this to a point with a little table where you have the alternatives in one dimension, and "desired characteristics" in the other; the table is then filled with rankings (e.g., scale of 1-5, red/yellow/green dots, whatever) of each alternative on each criterion. You've all seen these kinds of comparison tables, e.g., comparing pros/cons/features of various software products, etc. Again, it should be clear how those rankings could have emerged from the analysis that you did, versus just something you're making up. You then have a short paragraph that discusses the table, what it shows, and how this leads you to choose alternative X as the most promising solution to that technological design decision.

- ☐ **Proving feasibility:** So you've done some basic investigation and analysis and have made a rational choice of one particular alternative as the way to go. Moving forward, however, you'll want to go beyond this preliminary analysis to actually show that the solution will work for the challenge raised by this particular project, i.e., when used for its actual task on the project. So in this short closing paragraph, you outline initial testing and your plans for further testing/validating your choice. In particular, you outline some specific demos that you tried and how they will develop as part of the "Technology Demo" assignment due later in the first Capstone semester.

- ☐ **Technology Integration:** In this section, you need to bring it all together. You've introduced individual challenges and how you plan to solve them, but how will all of these "micro-solutions" come together into a coherent overall system? A great way to organize this is to write a little intro where you segue in, then talk about needing to put all the pieces together into a coherent architecture that is capable of satisfying all of the product requirements. You then introduce a system diagram of your envisioned system that shows how the major elements relate to each other. What things are connected to others, what things are inside or part of others, and how data or tasks flow between them? Then you briefly walk through the diagram in the subsequent narrative, making it clear

how the parts work together within the broader product you're building. This is essentially a first cut at envisioning an overall software architecture for your system.

- [ ] **Conclusion:** What you're trying to do in any conclusion is to (a) remind the reader of the problem and how important it is; (b) summarize what you've covered in the document, including a few highlights; and (c) close in conveying the sense that you've done a complete and competent job, and how you'll now be moving forward with the projects next steps. Keep in mind that many busy readers (e.g., your boss/division manager/CEO) will read just two things: the document intro and the conclusion.