SE577 Software Architecture Course Project

Overview
This class will have an overall course project broken down into multiple deliverables. You will be playing the role of a senior software architect influencing and controlling the definition of the overall architecture for a new product or service from the time it was ideated right up to the time construction starts. Along the way we will iterate through various deliverables and end up with an architecture description of a proposed system to be built following a modern and realistic process.

For this project you will be steering a series of architecture deliverables that will ultimately result in the project being funded and constructed. You can come up with any idea that you want for the proposed product or service that you will be architecting. It can be a new online service, a game, a product, basically anything you want. If you get stuck you can always fallback to creating the next great "e-commerce widget for sale platform". The requirement for the project is that it has a modern realistic architecture and must:

- Have at least one integration with an external service. You can treat that service as a "black box" but there must be at least one integration. Some examples are SAP for inventory management, Salesforce for customer management, Google for mapping, Stripe for credit card payment, Twillio for SMS/Video, etc.
- Be robust enough to support the definition of at least 10 major architectural components. Sorry no stand-alone tic-tac-toe systems.
- Must have a clearly defined data tier, along with clarity on key architectural choices being made to support the data tier (e.g., eventually consistent vs atomic, SQL vs NO-SQL)
- Must address at least the resiliency, scale and security architecture quality attributes. Likely you will incorporate others, but these 3 are the minimum.

Deliverable 1 – Motivate the need/value for the product you will be creating – Start of the "Working Backwards" approach
For this deliverable we will be following the Amazon "Working Backwards" approach. First start by doing some independent learning/research on the "Working Backwards" approach. There are many free resources on the web, and even an entire book (with the same title) dedicated to this practice. The main benefit of this approach is that many products are created with an internal view around what an organization thinks customers need or want. Once built, they under-deliver on customer expectations. This approach flips the paradigm taking a customer viewpoint to motivate the need for creating a product that is exciting. For this deliverable you will author:

- A draft press release. This is a key differentiator of the working backwards approach. You basically "pretend" that the product has already been built (before anything has actually been accomplished) and your are creating a press release about the product to

engage the media and excite potential customers. This **one page** document would be formatted like a real press release (google what they look like) announcing a hypothetical new product or service offering that you are proposing. The press release would focus on customer benefits, key features, and the value proposition of the proposed offering. It would be written as if the product or service were already completed and ready for launch.

- A draft FAQ (frequently-asked-question) document. This is another key part of this process where anticipated questions about the product or service you are proposing would be elaborated with accompanying answers. You should develop a total of six FAQs, 3 of them as "external" FAQ, and 3 of them as "internal" FAQ. External FAQs target questions that potential customers, or the media might have about the new offering. Internal FAQs are questions that internal stakeholders, think CIO, CEO, Executives, etc might have about the product/service.

WHAT TO HAND IN: A single PDF with the one page Press Release, followed by an additional page or 2 (shorter is better) with the 6 FAQs in the form of both questions and answers.

Deliverable 2 – Development of 5 important Architecture Decision Records for your new proposed product or service.
For this deliverable you will author a series of ADRs based on what you think are the 5 most important decisions that must be made to shape the software architecture. ADRs are used to gain consensus given defining software architecture involves trade-offs. You can pick a basic ADR template off the web (e.g., https://github.com/pmerson/ADR-template/blob/master/ADR-template.md) for this deliverable. The main goal of an ADR is that it starts off as clearly identifying an important architectural decision that must be made, along with the context for why that decision is important. As the ADR process evolves, several candidate decisions are proposed, along with the consequences (both positive and negative) associated with each option. Ultimately a final decision is made and documented.

WHAT TO HAND IN: Your 5 draft ADRs. These may be submitted as a PDF containing all 5, or even better yet, a link to a GitHub / GitLab repo where each ADR is authored in markdown as ideally ADRs get managed like code.

Deliverable 3 – Develop an architecture model (one or more views) to document the solution architecture for your product and service. You can use UML, Lines/Boxes, or something like C4 to develop this model. The key aspect of this deliverable is to define the overall architectural components of your proposed solution, along with some high-level technical details. In addition to the visual model, create a short summary highlighting important aspects associated with your overall solution architecture.

WHAT TO HAND IN: A pdf that includes a visual model showcasing your proposed software architecture along with text to complement the visual model. The text can be some concise paragraphs mixed in with the visual models, or the visual models can be annotated with callouts (e.g., a circle with a number in it) that then are described in a table format.

Deliverable 4 – Create a 5 minute video presentation where you present the proposed solution you created in deliverable 3. An effective architect must have the ability to clearly communicate the technical vision for the architecture to a variety of stakeholders. For this video you can assume you are presenting to a technical audience. You can record this anyway that you want, but the recording must be internet accessible (you cant submit video files via blackboard). That said, the videos do not have to be publicly accessible and can be deleted after the course. Zoom recordings, Youtube posts are all fine and can be password protected or locked down.

WHAT TO HAND IN: A URL link to your final 5-minute video presentation. Please double check the links accessibility prior to submitting this part, include in the blackboard submission any additional information necessary to access the video such as the video's password.