Hello!

Part 1

After skimming through most of the assigned reading from the book, it seems to me that the design process for the system engineering method is one of the most important contributors to a successful project. I believe this is true from my own development experience as well. While going through the course material, I'm thinking a lot about the projects I've been on and where the pain points were in the system engineering process. I think really what it boils down to is inadequate design practices!

A lot of projects tend to want to follow "Agile" principles of iterative development and design and are afraid to do any form of top-down, waterfall-esque design (even at the front-end definition of customer need and a system development road map). I've noticed that this adds a lot of re-work, re-design, and back tracking in development. A lot of people who are major proponents of this methodology seem to think that this is the only way to be able to change direction quickly (or with agility, hence the name "Agile") in response to changes in stakeholder requirements. It's clear to me that leadership in my pervious projects has been afraid of the conceptual and preliminary design process being too far reaching, where it should just be a high-level outlining of needed technologies and high-level event storming with the client. Instead they over-correct and don't do enough of the top level design. These steps are crucial to understanding staffing requirements, needed technologies and tools, and optimal staff structures (team composition and skill distribution).

Something that I don't find important, for software development in particular, is staff location. With more engineers getting used to remote development environments, developers being within the same geographical location as the client doesn't seem to be as important anymore. There are several tools that help facilitate working from home, including virtual offices and video conferencing software. Where this *does* become a concern is with government contracts. A lot of government contracts require that all staff be physically located within the United States. Another consideration is staff being in different time zones. A lot of the projects I've been on expected all engineers to work within the 9-5 of their corporate office, and it was understood that individuals working for that company would work on the company's time rather than their own.

Another piece to this process that I think is highly important is the decision-making process. This includes electing staff and titles on the project that have decision-making power. I've noticed that projects tend to run more smoothly when the masses aren't trying to make decisions, but rather there is a central individual or group of individuals that give

software development design direction. This is my own qualm with flat corporate structures that operate under the guise of all individuals having the same decision-making power. In my experience this just leads to a lot of chatter and confusion.

Part 2

I don't think I have any proposed changes to the model. If I had one area to focus, I would go into detail on the design process. Here I separated out the conception and preliminary design process into the System Engineering model, and emphasized the iterative refinement process as a sub-heading of the design process:

1. Title page
2. Executive Summary
3. Document History
4. Table of Contents
    1. Introduction
    2. Purpose
    3. Document Overview
    4. System Overview
    5. Project Schedule
    6. References
5. System Engineering Processes
    1. Project Organization
    2. Environments
    3. Decision-Making Process
    4. System Engineering Model
        1. **Method for Compilation of Client Needs**
        2. **Client Driven System Road Map**
    5. System Engineering Processes
        1. Configuration Management
        2. Requirements Engineering
        3. Functional Analysis
        4. Design Processes
            - **Iterative Design & Agile Refinement**
        5. Development Processes
            - Software
            - Hardware
            - System Integration
            - Build Management
    6. Verification
    7. Validation
6. Specialty Engineering

7. System Deployment
    1. Site preparation
    2. System installation
    3. System checkout
    4. User training
    5. Support engineer training
8. Product Support
    1. Maintenance
    2. Logistics support
    3. Disposal

References:

Blanchard, B. S., & Fabrycky, W. J. (2016). Systems Engineering and Analysis (5th ed.). Pearson Learning Solutions. https://coloradotech.vitalsource.com/books/9781323417522