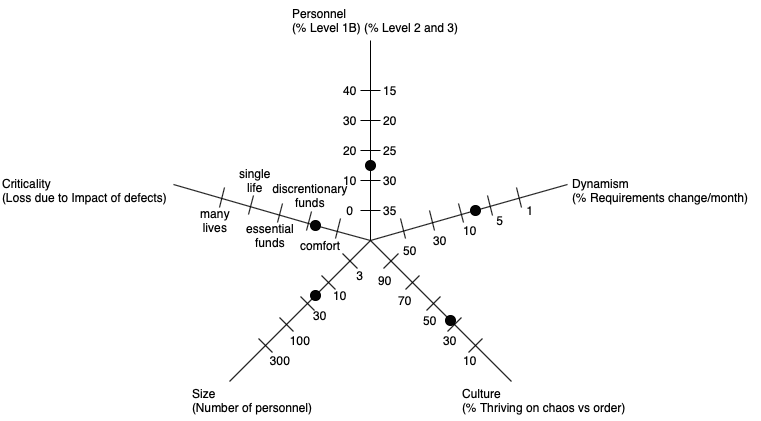
# Intended Software Engineering Process

## Spider Diagram

The below spider diagram illustrates the team’s thoughts on Emxsys’s current company outlook and its software systems, as well as the team’s belief in the project and its necessary features needed for development.



### Personnel

Since Emxsys uses the Campbell Prediction System, the basic algorithms are already implemented. Thus, the features that would need to be implemented are fairly trivial, which makes the need for top-tier employees much less than having to reimplement the algorithms.

### Dynamism

The project the team is trying to implement is well documented in the previous section, and it would only need relatively small modifications as the project is being developed.

### Culture

There are no defined culture standards for Emxsys since it is open sourced currently, and the nature of the platform will cause hesitation to make dramatic changes. So, the culture would tend to be more orderly rather than chaotic.

### Size

Emxsys is open-sourced, and it does not have paid employees constantly updating the system. Also, the features that are suggested by the team are numerous and quite sizable. Thus, the number of personnel needed would need to be increased semi-dramatically.

### Criticality

While the nature of the system deals with wildfires and many lives are at stake, the algorithms to predict the wildfires are already in place. The solution the team is proposing will be focused mainly on customer service and geographical analytic systems. Therefore, the criticality is closer to discretionary funds, rather than being towards the higher end of the criticality spectrum.

## Project Management

Project management is and will be an ongoing process throughout the SDLC for the team. However, an initial outlook on the team’s plan is detailed below.

### Release Plan

The current release plan is uncertain. However, through domain analysis and gathering of various information, a clearer picture will develop, in which, a rough estimation of an inceptive release plan will be developed. As the project gathers traction and the most necessary features are decided upon by the team, these will be prioritized in order to complete a minimum viable product (MVP), with the less important features as an extension for a more complete product.

### Iteration Plan

Once the release plan has been developed, it will be divided into various iterations. In each iteration, a comprehensible list of user stories, use cases, functional and nonfunctional requirements will be detailed.

### Risk Plan

As with any endeavor, there are numerous risks that will occur throughout its lifetime. The team will attempt to stay ahead of these issues by outlining potential risks, how each might occur, the necessary mitigation steps, the proper way of detailing new risks, and how to document the completion of these issues.

### Linear and Parametric Estimation

The team will use Scrum methodology to assist in breaking down the project. The use of a product backlog will allow the team to get an overview of all the features that will need to be completed. The product backlog will be broken up by numerous sprints categorized by major topics. Each item in the product backlog (and sprint backlogs) will be given a value based on difficulty. The values will be: 1, 2, 3, 5, 8, 13. Using values based on difficulty rather than hours will provide a much more accurate estimation so that the sprints can be accurately distributed and planned for.

### Project Schedule

The project schedule will be a visual representation of a timeline, and it will be created following the completion of the iteration plan. This will allow the team to have an accurate and encompassing breadth of the project. Therefore, the sprints can be distributed evenly, and it will allow the team to understand how long the project will take to deliver a proper project schedule.

## Requirements

The team will need to gather all necessary requirements and information regarding the project to give the following work products real meaning. Once this is done, the team will document, detail, and analyze which are of the utmost importance to the project.

### Storyboards

Storyboards will be used in order to give a visual representation of the system. This will provide aid to developers and other employees, and to future customers who will be using the team’s product.

### Problem Statement

The problem statement will give insight into exactly what the team is attempting to accomplish with the project. This will outline major goals, minor goals, future extension points of the project, and any general information regarding the topic.

### Business Case

A business case will be completed in order to outline if the project is possible due to a number of various constraints. It will be broken down by these constraints and outline how each will be completed or if the area is even possible given the current standing of the team.

### Use Cases

Use cases will provide a very generic, yet informative, outlook into the system and how users will interact with the system. From these use cases, a use case diagram will be created to show a concise description of all these use cases.

### User Stories

Each use case has a number of user stories. These are important and descriptive points that will need to be developed in order to have a successful and encompassing completion of the project.

### Scenarios

From the use cases, scenarios will arise to detail how each aspect will flow for the system. This is significant as it ensures total completion of the use case diagram.

### Nonfunctional Requirements

Nonfunctional requirements will need to be developed to ensure the system has the needed accessibility, reliability, security, usability, safety, testability, maintainability, extensibility, and scalability. This will ensure there are no future issues regarding any of these topics so that the system will be ready for commercial use.

### Acceptance Plan

In order to have a sense of how far the system is progressing, the team will need to gather success criteria, and, from this, develop and agree upon an acceptance plan. This will ensure no pieces are left out and detail how successful the team was following the completion of the project.

### System Tests

The team will provide details on how to properly test the system. In all testing scenarios, the goal is to break the system or find any bug that might arise. This will ensure the system is complete with no holes in it.

## Analysis

The team will need to provide detailed analyses for various aspects of the system. In all of these analyses, the team will attempt to give a better understanding of each area for the team’s benefit as well as any other user of the system.

### Domain Analysis

In the domain analysis, the team will gather all information surrounding the area the system is attempting to solve. In the case of the team’s project, this will detail areas surrounding weather, wildfires, customer service tools, geographical analytics, etc.

### Problem Analysis

The problem analysis is an investigative analysis into the problem the team is attempting to solve. It will lay out issues with the current competitive environment in order to give insight into areas of improvements to the system, processes, procedures, and designs.

### Solution Analysis

The solution analysis will be an examination of the problem analysis in order to analyze where the major flaws are so that the team is accurately trying to solve the current problem. It will provide details on the ongoing issues and the steps to fix these affairs.

### Sequence Diagrams

The team will construct sequence diagrams in order to give a chronological picture of how users will interact with the system and how the system will handle these actions.

### Collaboration Diagrams

Collaboration diagrams will be used in order to provide an illustration of how the team is set up and how it will interact in certain scenarios.

## Architecture

The system architecture is one of the most significant portions of any system. It will provide insight to what the system will be, and how it will be implemented. The architecture will be a roadmap to the team to ensure the team is correctly building the system.

### Subsystem Model

The subsystem model will show all of the underlying systems that will aid in completion of all requirements. Each subsystem will have all of the necessary documentation and diagrams to show it works, describe its features, and detail its importance to the overarching system.

### Target Environment

The target environment describes which layer of architecture the systems will operate. This could be on the user’s hardware, the system software itself, or the cloud for data retrieval.

### System Architecture

The system architecture is a conceptual model that defines the structure, behavior, and views of the system. It will provide a formal description and representation of the system that is organized logically to show reasoning and importance of each subsystem.