EECS 448: Project 2 Design Paradigm Paper
Team 7 3/12/2021

The team who previously worked on this project likely used an Object-Oriented design paradigm as the core of their design paradigm while utilizing some of the design methodologies associated with Top-Down Functional Decomposition and Component-Level design. This assertion is made based off of the nature of the files this team received. The legacy code was broken into components that would generally be associated with the building blocks of a game, such as a Gameboard class, a Player class, and a Ship class. The way these components were defined strongly suggests that the team started by using Top-Down Functional Decomposition to determine how to divide the project into components. From there, each component of the legacy code contains clearly defined class variables and methods. This strongly implies that the previous team then began to utilize Component-Level design methodologies for defining the nuances and specific tasks each of component classes would handle.

At the core of the previous team's design paradigm was Object-Oriented design. A hallmark of Object-Oriented design is the use of classes, with each class having certain specific attributes and functions assigned to them. In the case of the previous team, they divided the core components of their game into specific classes. Some of these classes include "board.js", "player.js", and "battleship.js". Each of the aforementioned classes handle some specific task integral to game functionality. For example, the battleship.js class's job is to generate the game board that the game takes place on. It accomplishes this through utilizing the function "generateBoard()".

From the observations this team made, we can safely conclude that the team which previously worked on this project utilized an Object-Oriented design paradigm as well as some design methodologies from the Top-Down Functional Decomposition and Component-Level design paradigms.