

TECHNICAL DOCUMENTATION

OOP IMPLEMENTATION GUIDE:

Abstraction

- Abstraction is used in hiding complex systems. It is implemented on Game.php, GameManager.php, and config.php by using private functions methods that outside .php cannot call like private function initializeGame() { ... }, private function loadFromSession() { ... }, private function saveToSession() { ... }, etc. in Game.php. They also hide loading and saving session data, checking if ships can be placed, reset logic, initializing game state. The public only sees the results, not the process. Abstraction is also implemented on public function methods as they only reveal what the users should control like public function placeShip(...), public function confirmPlacement(...), public function attack(...), etc. in Game.php. They provide simple actions like attack, place ship, reset, without exposing internal complexity so the user doesn't need to understand how the board updates or how ships are stored

Encapsulation

- Encapsulation is used by hiding data inside classes and allowing controlled access through methods (getters, setters). It is implemented in class properties of each class like private \$name;, private \$board;, private \$shipsPlaced = false; in Player Class in Player.php, same goes to other class that also have class properties in different .php. Interaction happens through public function methods

Inheritance

- Inheritance is used to extend parent classes' methods into their child class that is modifiable and is reusable. It is implemented in PHP codes for each ship that is located on the ships folder inside the zip file. Example of this is class Cruiser extends Ship, class Carrier extends Ship, and more using Class Ship as their base class or parent class.

Polymorphism

- Polymorphism is used to share the same method name of various classes but each behave differently. An example of this are the, for Cruiser Class, public function getType() return 'cruiser', and other is for Battleship Class, public function getType() return 'battleship'. Method from parent class is overridden in the child class.

SETUP INSTRUCTIONS:

1. Extract the zip file into “xampp\htdocs”
2. Open xampp and start apache (Make sure the apache directory is same as the directory of your xampp file)
3. Go to google and input “localhost/gridstrike/” (Make sure the filename is same as the name of the folder which is “gridstrike”)
4. Click start to run the game

GAME RULES:

- Each player should place all available ships before proceeding by clicking the type of ship first then clicking a position inside the grid
- Player can place their ships only either horizontally or vertically in any part inside the grid
- The game is turn-based and each player gets only 1 strike each turn. The one that has a red outline in their grid means that they are the one being attacked. If a player strikes a hidden ship, it will mark as a red grid. If not, it will mark as gray grid
- The winner will be determined by who sunk all the ships first