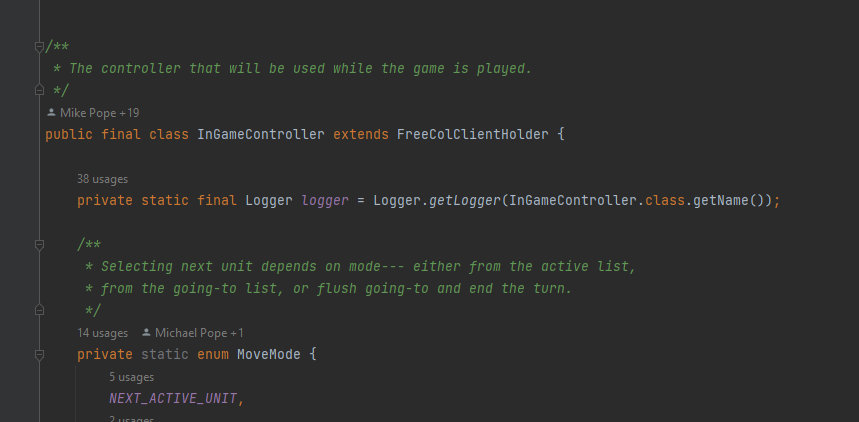
* Code Smells

* Large Class, InGameController

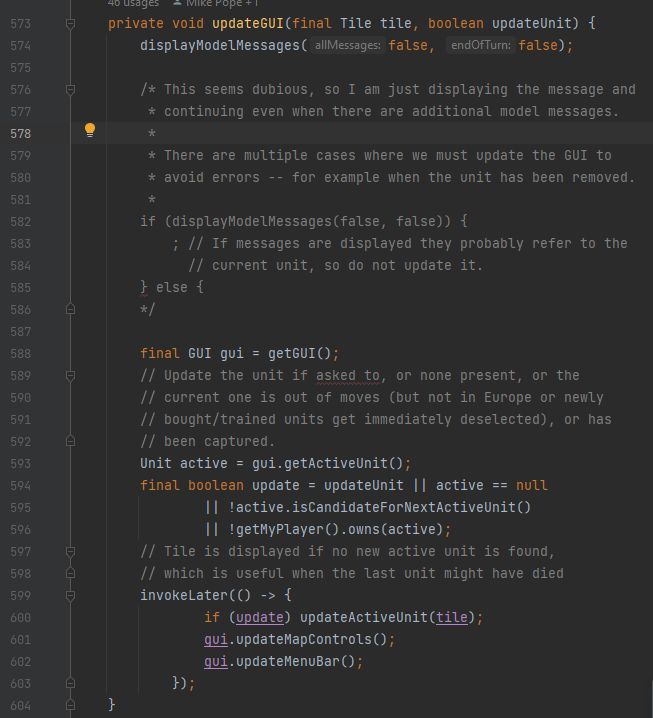


A classe *InGameController* como um todo representa um **Large Class** com 5385 linhas.

A classe é responsável por uma grande parte do funcionamento do jogo, tornando-se difícil de manter e compreender.

Como solução poderíamos dividir as funcionalidades desta classe em classes menores especificá-las, cada uma com uma responsabilidade única, facilitando a sua compreensão e manutenção.

* Feature Envy, updateGUI()

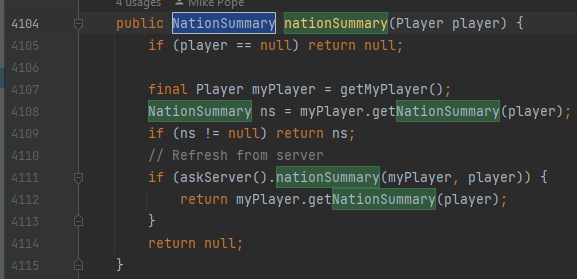


O método updateGUI() faz muitas chamadas e manipulações a objetos do tipo GUI.

O método está nas linhas 573-604 da classe InGameController e depende da classe GUI dado que realiza várias chamadas para atualizar os elementos específicos da interface do usuário, podendo indicar um acoplamento forte e perturbar a baixa dependência entre classes.

Uma solução seria reduzir a dependência do método em relação à classe GUI ao mover algumas responsabilidades relacionadas à interface do usuário para métodos da própria classe GUI ou criando novas classes para partilhar as suas responsabilidades.

* Feature Envy, nationSummary()



Linhas 4104-4115 da classe InGameController.

O método usa grande parte dos dados de objetos do tipo NationSummary e Player, causando **Feature Envy** por depender demasiado da classe.

Seria uma boa resposta para este problema mover o código relacionado com o NationSummary para dentro da classe Player ou oferecer um método mais geral sem pesar numa outra classe.