Documentation for Chess Engine

**TODO:**

1. **Look at and decide on Visual Studio C++ Build Tools**
2. **Look at and decide on Chess GUI Frameworks**
3. **Read up on Circle CI, and other Chess engines**
4. **Start on Design Doc/ Algorithm**

**Intended Design and Philosophy**

The intent of this Chess Engine is to play based on only 1 node level. The calculation will be only based on the current available moves. Thus, the Chess Engine will not take into consideration 2 moves and beyond. The intention of this chess engine to develop a set of chess principles that could hopefully determine the course of play. These principles will determine the assessment of the available moveset for the current position (only).   
  
Why I want to develop this Chess Engine: To see the potential/limits of something a primitive evaluation. If chess engines are determined through the depth of the move tree, I want to explore the other direction.

Potential Difficulties/Points to consider:

1. It may take quite a while before a good set of principles are developed
2. The principles will require a lot of fine tuning
3. The principles/model could also be wrong

**Thoughts/Questions about approach:**

Should there be machine learning? To develop the model based on the principles could be quite hard to do based on my understanding.

How do I arrive on the set of principles?

Algorithm:

**Frameworks/ Chess GUI to be used:**

**Possible Tools to be used:**

Circle CI/ (Or something similar to Travis CI/ Free)

Component Design:

Architecture:

Testing:

SDLC:

Possible Diagrams:

Sequence of Development: