

SP-14 Blue Chess AI

CS 4850, Fall, 2025

Project Team

Roles	Name	Role	
Team leader	Conner Handley	Programmer, Tester	
Team members	Lowell Luckey	Programmer, Tester	
	Ryan Nguyen	Documentation, Tester	
	Ulises Villagomez	Documentation, Tester	
Advisor / Instructor	Sharon Perry	Facilitate project progress; advise on project planning and management.	



1.0 Project Overview / Abstract (Research)

Using Javascript\HTML, a chess engine will be created that will allow player vs player, and player vs AI. The chess AI opponent will evaluate the board using search algorithms, i.e minimax, to achieve the best possible move against the player. The chess engine will check for legal and pseudo-legal moves, check, checkmate, and stalemate, and other special moves such as castling. The chess engine will be integrated into a website that will hold our project information.

2.0 Project website

<https://14blue-sp.github.io/website/>

Deliverables

- Project plan to organize and outline the project's development timeline
- Software Requirements Document (SRS) and Software Design Document (SDD)
- STP & Development Document
- Prototype Presentation
- Final Report Package that includes a final report, source code, a website and a final presentation video
- C-Day Application/Submission

Group Meeting Schedule Date/Time

- 9/13/25 Saturday @ 7pm
- 9/26/25 Friday @ 7pm
- 10/10/25 Friday @ 7pm
- 10/28/25 Friday @ 7pm
- 11/19/25 Friday @ 7pm

Collaboration and Communication Plan

Communications and collaboration will take place daily throughout the project's timeline via discord and cell as needed. Online meetings occur at least once a week and any additional meetings may occur as needed. In person meetings will occur periodically as deemed necessary.

Project Schedule and Task Planning

Potential Limitations

In case the Chess engine is too strenuous to run through Javascript, our group has planned on separating the chess engine from the website and leaving the website to only hold our project information.

Version Control Plan

This version control plan outlines a strategy for managing source code, fixes, and releases to ensure there is consistency, traceability, and collaboration throughout the development life cycle of the Chess AI project. It includes:

Repository structure -

A main directory: *chess-ai*

Platform hosting: GitHub

Directory structure following conventional naming semantics for web development:
main (operations), index (structure) , style (design), game (game state)

Branching strategy -

A main/master approach where the main branch is a stable product always ready for deployment. Temporary branches are created for bug fixes, new features, and testing that are isolated from the main branch maintaining stability.

Workflow -

Contributors pull requests and commits only after 1-2 peer reviews. Branch merging follows the same peer review requirements.

Model Versioning -

Utilize semantic numbering and naming schemes for version updates. Utilize appropriate tags for major updates.

Backup and recovery-

Mirror cloning required at every version update. Potential for additional mirror cloning when required.