P3 Presentation

DedicatedRAMs Team

Ethan Liem

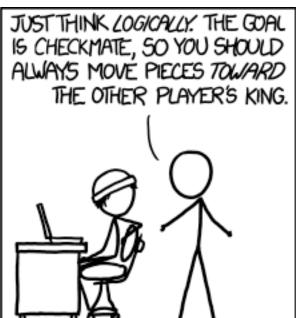
Cathy Lee

Najy Faour

Hannah VanderHoeven

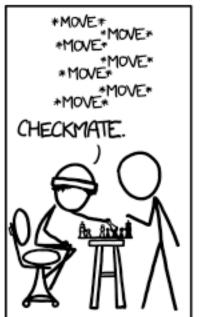
Axel Wahlstrom











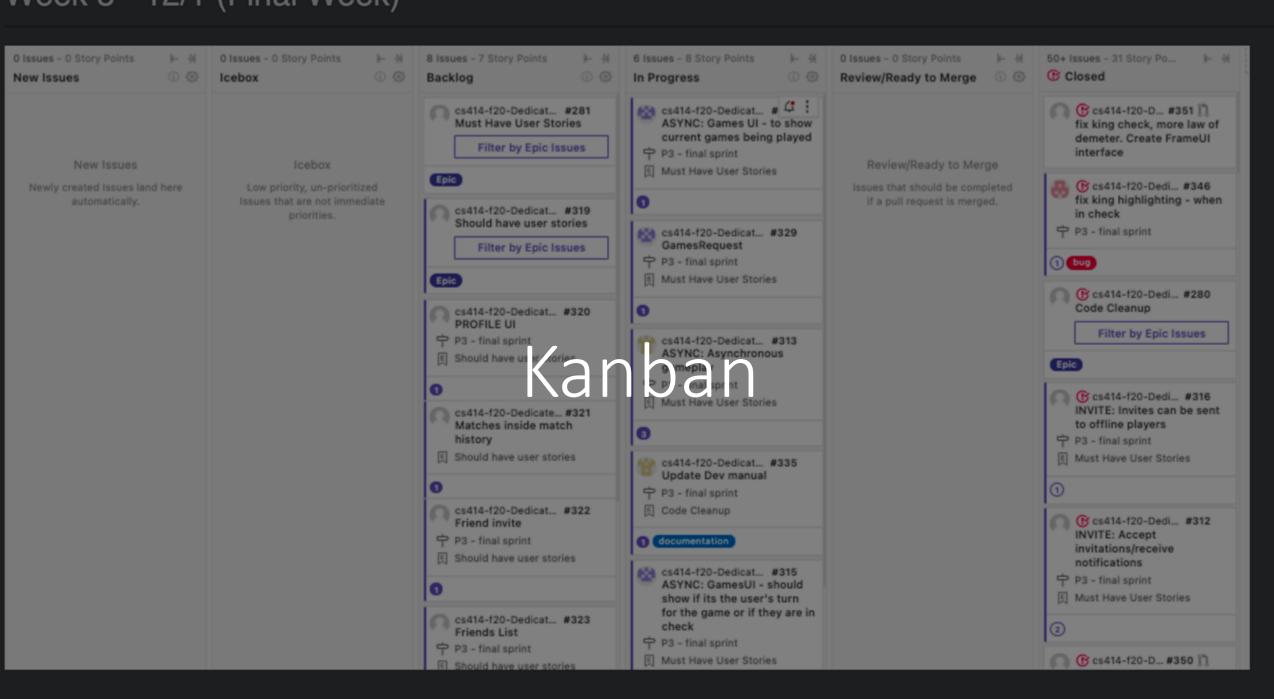






Design

Row: Tasks, Column: Classes	ChessBoard	ChessPiece	Knight	Bishop	Rook	Queen	King	Pawn	Game	Player	Vest	Move	Piece Movement	Move History	GameUI
95. Create ChessPiece Class		Х													
105. Create ChessBoard Class	X														
118. ChessBoard toString()	X														
96. Knight Class			Х										X		
97. Bishop Class				Х									X		
99. Rook Class					Х								X		
102. Queen Class						Х							X		
100. King Class							Χ						X		
104. Implement Castling	X						Χ						X	X	
98. Pawn Class								X					X		
103. Pawn En Passant								X					X	X	
108. Game Check Logic	Х						Χ			X				Х	
121. Implement Player Class					1 .			ı		Х		•			
107. Implement Plundering	X	ac		\bigcirc XI	X	-\ /×	Х			\	つ †	r	V		
109. Game Checkmate Logic	X	ac	こロ	$ \mathcal{O} $		LY			X	Vk	aι	.	lack	Х	
132. Implement Vest Class		Х									Х				
150. Implement Movement Class			X	Х	Х	Х	Χ	X				Х	X		
123. Game Draw Logic	X													X	
123. Pawn Upgrade	X							X							
107. Implement Game Class	X								Х						
121. Implement Player Class									Х	Х					
208. Pawn Promotion User Input	X														
224. Player With Class	X	Х	Х	Х	X	Х	Х	X	Х	Х			X	Х	Х
231. EnPassant UI	Х							X	Х				X		Х
232. Castling UI	Х						Х		Х						Х
204. Plunder UI - Movement	Х														Х
206. Draw/Checkmate UI									Х						Х
205. Check UI															Х



Summary

Sprint Output

Lee, Cathy

Date	Updates	Impediments
11-17-20	Delete User feature - account settings on start page	
11-19-20	Still working on delete user, modifying database accessor	
12-01-20	Delete User Account - Now removes games from players who delete accounts	
12-03-20	Updating design artifacts - TLM and Kanban Board	

Faour, Najy

Date	Updates	Impediments
11-17-20	Finished move and plunder coordination, started on saving game state.	
11-19-20	working on async, added save request/response	
12-01-20	Asynchronous	
12-03-20	Async - pretty close still need to load games	

Liem, Ethan

Date	Updates	Impediments
11-17-20	Code Cleanup, Fixed Notifications, Illegal Moves, Cleaned up game logic	
11-19-20	Bug fixes, work on class diagram and updated tlm	
12-01-20	Code cleanup, law of demeter, etc.	
12-03-20	Fixing Check	

Wahlstrom, Axel

Date	Updates	Impediments
11-17-20	Linked Client, ClientUI, and Server in UML Diagram, Began work on InviteUI	
11-19-20	Invite UI - Not present during this meeting	
12-01-20	Invite UI - Just need to populate tables with invites	
12-03-20	Invite UI - Final Features	

GameLogic Package

ChessBoard

Responsibilities Collaborators

- Player

- Vest

- Chessboard

- Movement

- Moves, captures, and plunders ChessPieces on the board
- Allows for the use of vests from plundering
- Initializes the starting positions of ChessPieces
- Keeps track of the board state (stalemate, check, checkmate)
- Promotes a pawn when it reaches the end of the board
- Knows which Player can move pieces and which King is in check based on player color.
- Contains a history of the moves that have been made.

- Game
- Player
- ChessPiece
- Move History - Vest

ChessPiece

Responsibilities Collaborators

- Knows its position on the ChessBoard
- Has a color based on the player.
- Knows whether it has moved or not.
- Belongs to a Chessboard
- Is either a rook, bishop, knight, pawn, queen, king
- Can gain a Vest of another piece Knows what ChessPiece's it can plunder
- Knows the Vest it has on
- Knows what legal moves it can make, in and out of Check

Piece Movement

Responsibilities Collaborators

- A collection of movements that a ChessPiece can make - ChessPiece

Vest

Responsibilities Collaborators

- Denotes which movement a ChessPiece has plundered based on a unique color.
- ChessPiece

- Knows the move a ChessPiece can make based on its vest.

- ChessBoard

Game

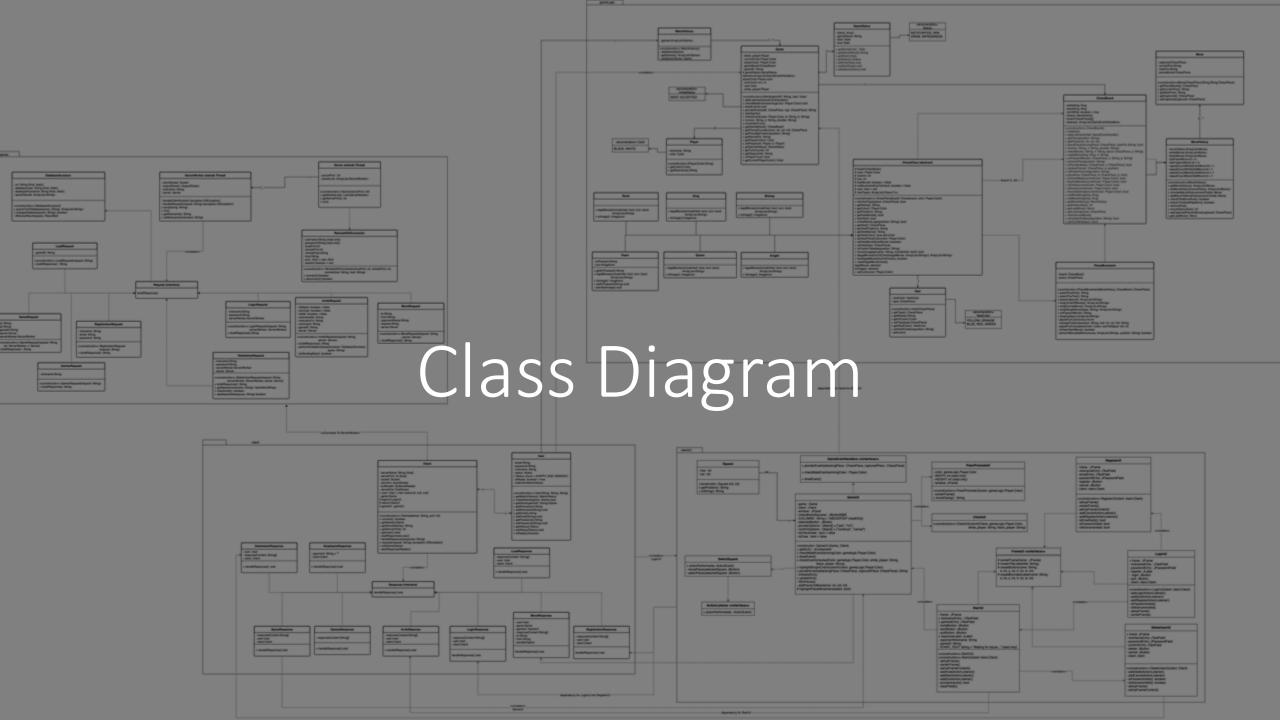
Responsibilities Collaborators

- Has two players and knows the current player. - User(s)
- Has a Game Status
- Has a chessboard
- Notifies players of plundering, checkmate and stalemate.
- Increments the turn
- Allows the player to move, capture and plunder based on player input
- Game History - ChessBoardÚl - Player

- Game Status

- Chessboard

CRC Cards



1. As a user, I can play plunder chess

Notes	 A user should receive dialog when they plunder a piece, and should receive dialog to use that plundered moss The game should tell you what legal moves a ChessPiece can make The board should look the same for both players (i.e. the player should always be the on the bottom of the game for both players) 					
Acceptance	e Criteria	 The game and its pieces follows the rules of Chess at Players are not allowed to move pieces illegally and A player should only be able to move a piece during A player should be notified when they can plunder a A game should end with a winner and a loser, or be at The game should notify users when a player is in che A game should also provide users with a GUI to play The GUI should include methods to get input from the The player should only receive a notification for chec A player should be able to plunder when using a plur The plundering player should only receive game prof Both players should receive prompts about checkma Only the player who is put in check should receive a A player should be notified when a piece has limited The king is highlighted when the opposing player is proposed to the player is proposed to the player is proposed to the player is player is proposed to the player is proposed to the player is player is player is player is player. 	illegal moves don't change the state of the game. their turn. move, and if they want to use that plundered move. a draw. eck. Plunder Chess e user to control the game logic kmate and nothing else when it happens adered move to plunder a piece. https://peac. https://pe			
Tasks 22 22 22 23 33	96. Knight C 97. Bishop C 98. Pawn Cla 99. Rook Cla 100. King Cl 102. Queen 103. Pawn - 105. Create 121. Create 128. Create 285. Separa 288. Law of 291. reimple 296. Ul notifi 289. Removi 294. Cleanu 297. reimple 298. Plunder 304. reimple	ChessPiece class lass and LegalMoves() Class and LegalMoves() En Passant ChessBoard Class Player Class Movement Class le client package into gamelogic and client demeter - board/move history ment addMoveToMoveHistory method less player when they can't move a piece due to check le turn order from chessboard lo class names ment simulate move method lul only one player is notified ment ChessBoard move Method chessboard tests levent handling removed from ChessBoard	77. Implement ChessPiece Plundering 104. King - Castling 107. Implement Games Class 108. Implement Checking 109. Implement Checkmate 124. Pawn - Upgrade 123. Implement Draw 132. Implement Vest Class 232. Castling - UI 231. EnPassant - UI 228. Add UI capturing and plunder dialogs 224. Connect Game with Player Class 217. Server on CS computer 208. Pawn Upgrade 207. Plunder - replace vest 206 Win lose draw - and checkmate 205. Check UI 204. Plunder - using movement 203. Plunder - vest and obtaining 202. ChessPiece - highlight possible moves 201. ChessPiece - movement 200. ChessBoard UI 146. Capture piece for En Passant Move 129. Move (request) 104. Implement Castling for the King 76. Add to UI the option to steal captured piece's moves (75. Add support for multiple games to the UI 48. Add basic UI for ChessBoard 49. Add basic UI for ChessPieces			

User Stories

Development Manual



DEMO IN VIDEO