

# Screenshots of taskboard Trello

## 3rd sprint

**Introduction**

**Documentation** Work in Progress

Class Diagram (See desc)

≡

**Documentation**

For project process, see the "Project Progress" board

1

**Fully Dressed Use Case Format**

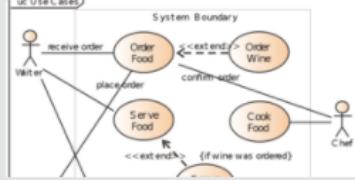
Use Case Name	WikiName, start with verb
Scope	System boundaries (corp, prog)
Level	Summary, subfunction, etc
Primary Actor	Primary system user
Stakeholders	Who cares and what they want
Preconditions	Must be true to start
Postconditions	What is guaranteed by success
Main Success Scenario	Typical: unconditional path scenario
Extensions	Alternative success or failure scenarios
Special Requirements	Related non-functional requirements (RAM)
Technology & Data	Varying IO methods and data formats
Frequency of Occurrence	Is this system used often
Miseries	Open issues, eg unmanageable failure scenarios

**Documentation**

Fully dressed user-case diagram

1

Make an application where the user can play chess. The application has a local multiplayer mode and vs AI mode. The AI has different difficulties. On the hardest, the AI will try to best the player



```

    graph TD
        Waiter -- "receive order" --> OrderFood
        OrderFood -- "place order" --> OrderWine
        OrderWine -- "confirm order" --> CookFood
        CookFood -- "if wine was ordered" --> Chef
        OrderFood -- "<< ext end >>" --> OrderWine
        OrderWine -- "<< ext end >>" --> CookFood
    
```

Legg til et kort...

**Technical demands**

A player should be able to play against an AI-player

The application should support vanilla chess rules

A player should be able to create a profile to check all past games

The application should have a highscore system/ranking system.

Pawn

Knight

Bishop

King

Queen

Rook

Board

Legg til et kort...

**Non-technical demands**

Multi-platform support

GUI

≡

Graphics should be open source

Source code and building scripts should be open source

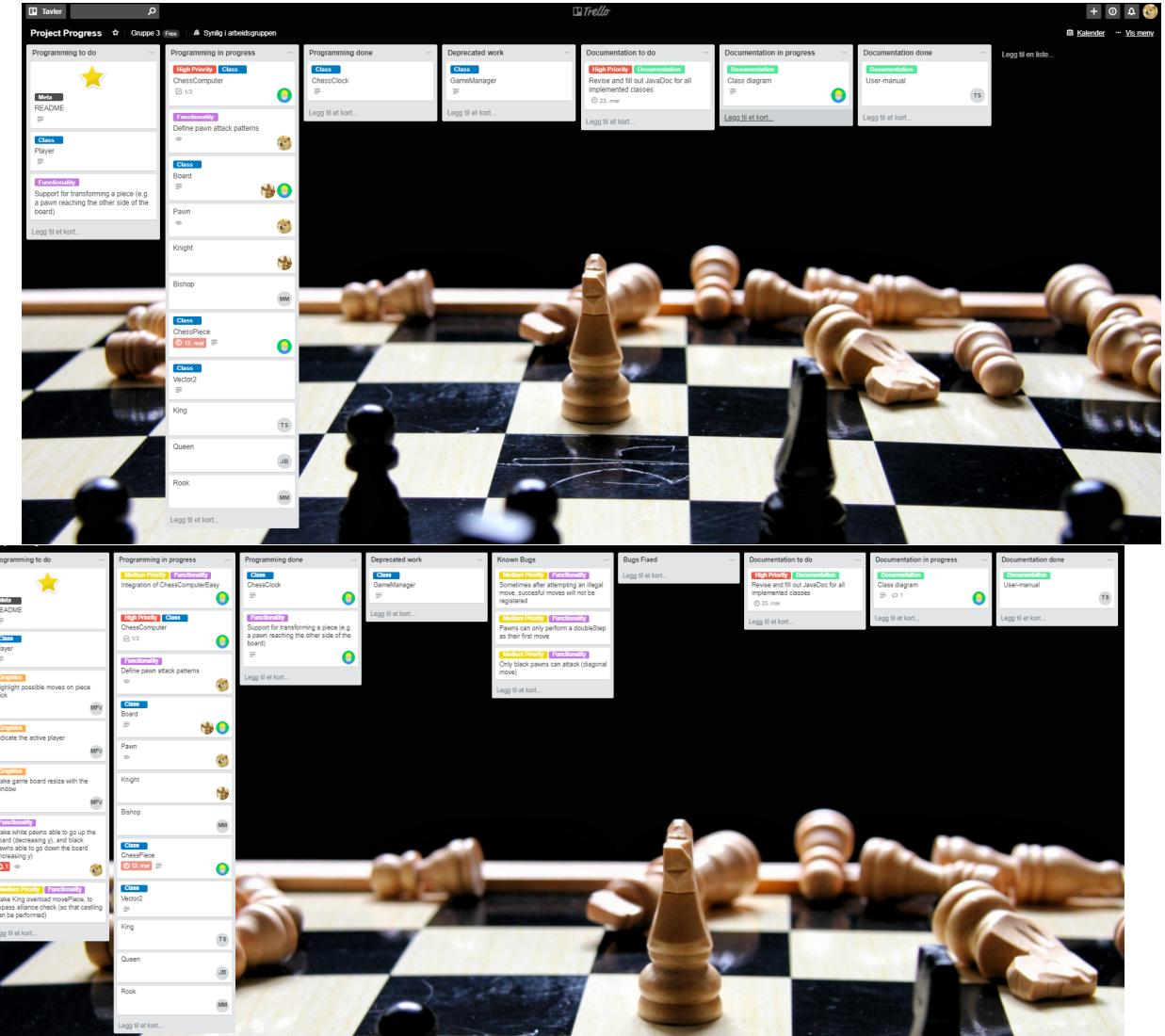
Java code should be documented in best practices using JavaDoc

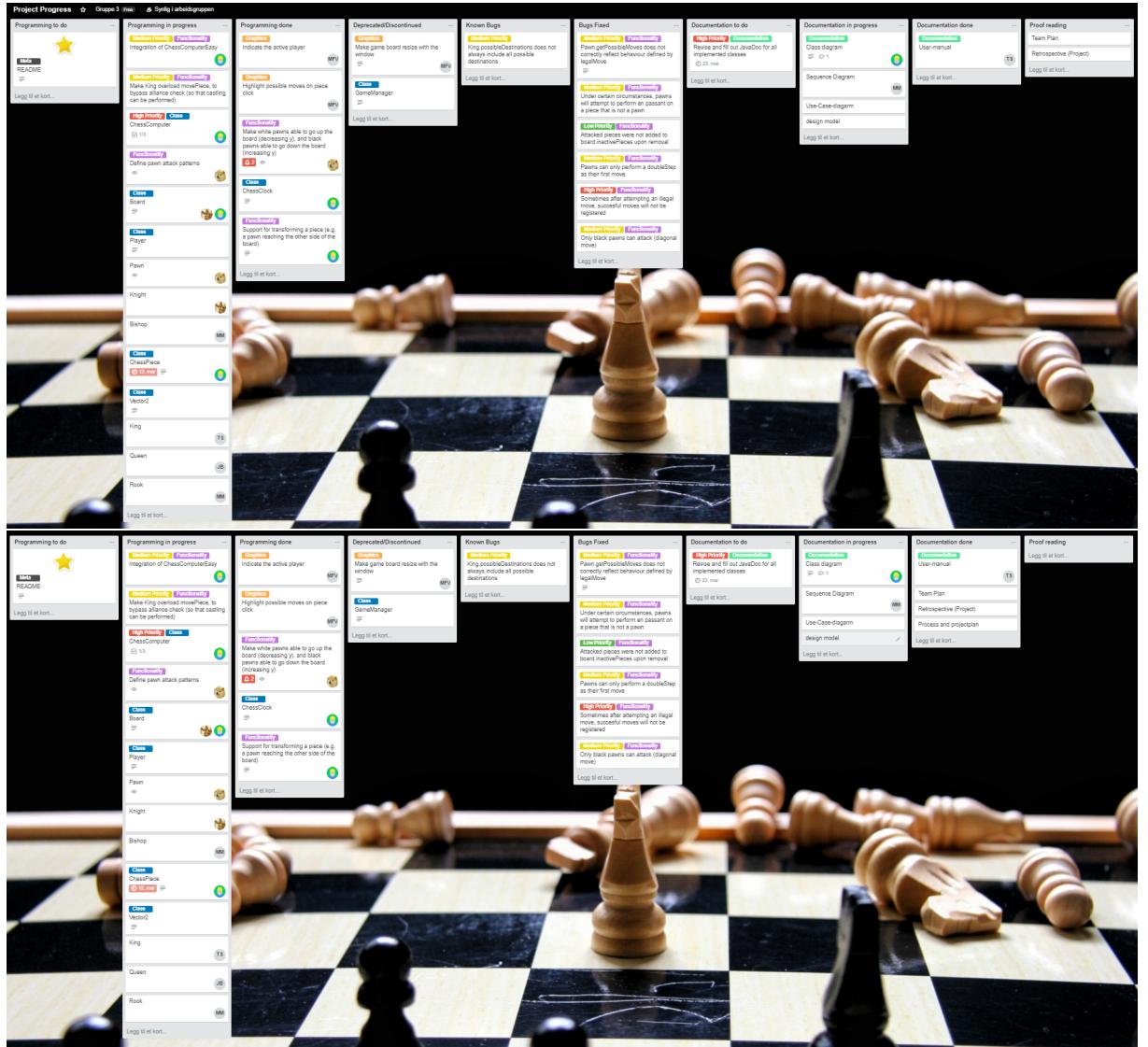
Easy AI-player should do its move in 1 second

Medium AI-player should do its move in 3 second

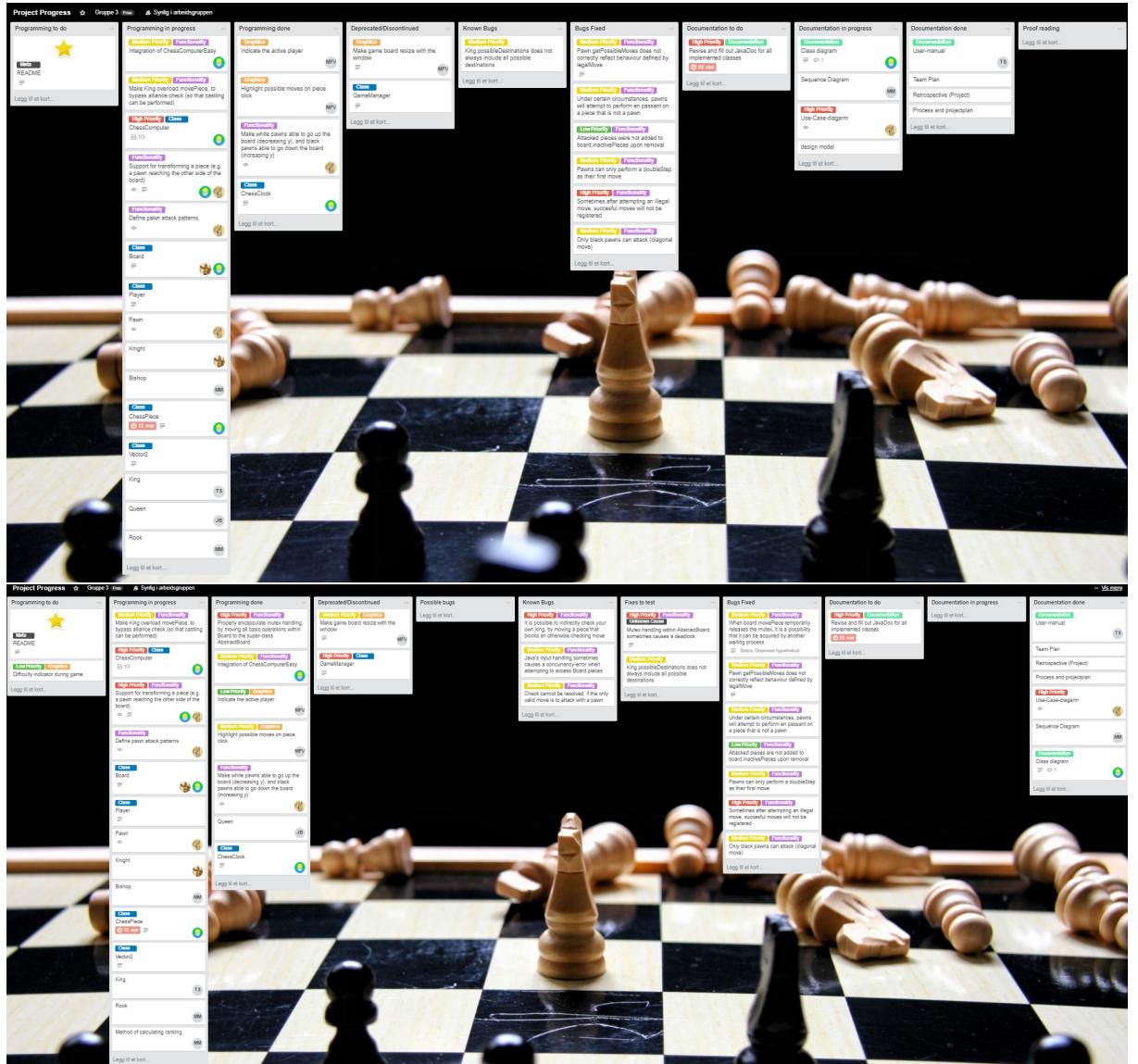
The application should be easy to expand to include different rule sets

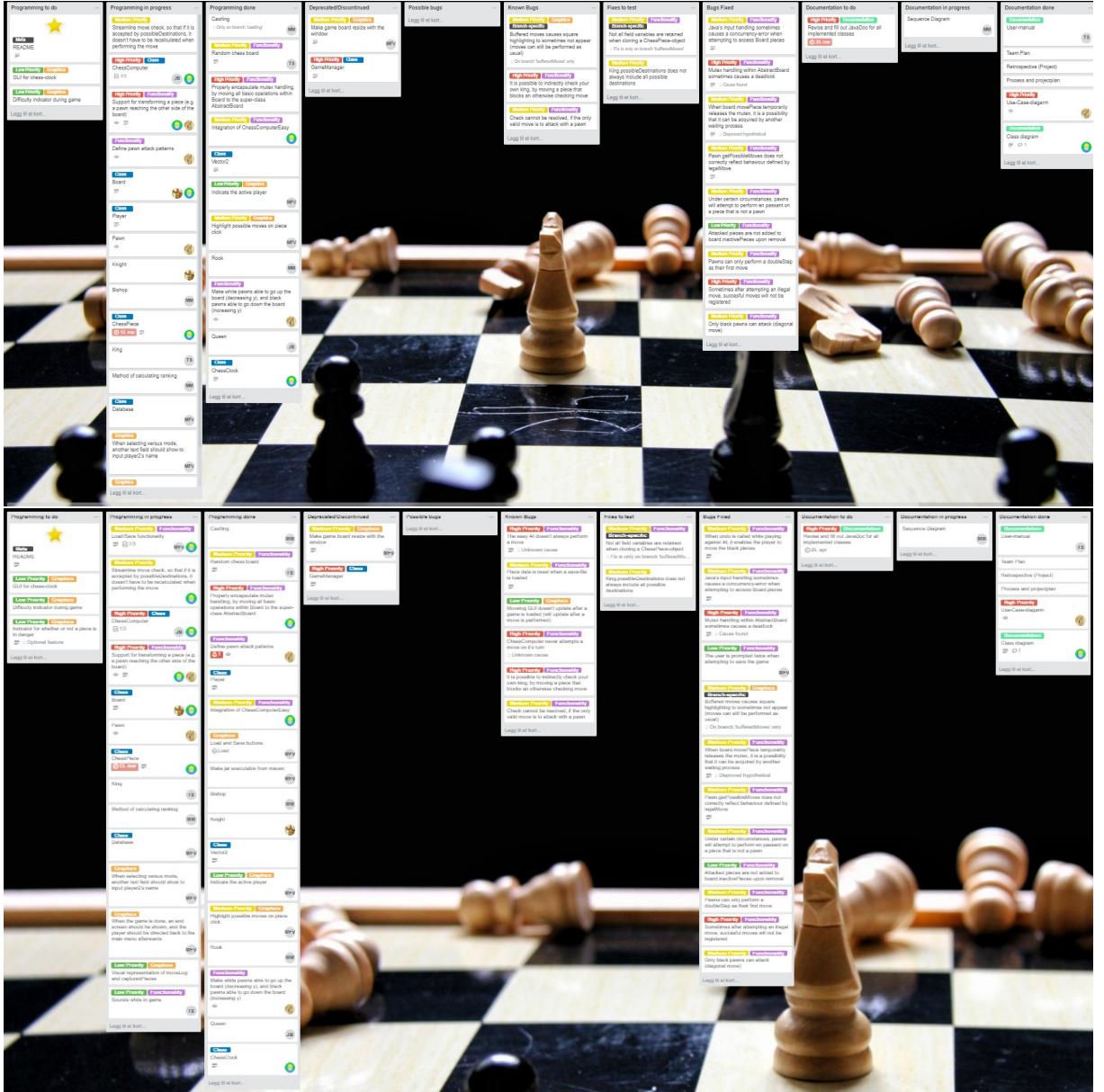
Legg til et kort...

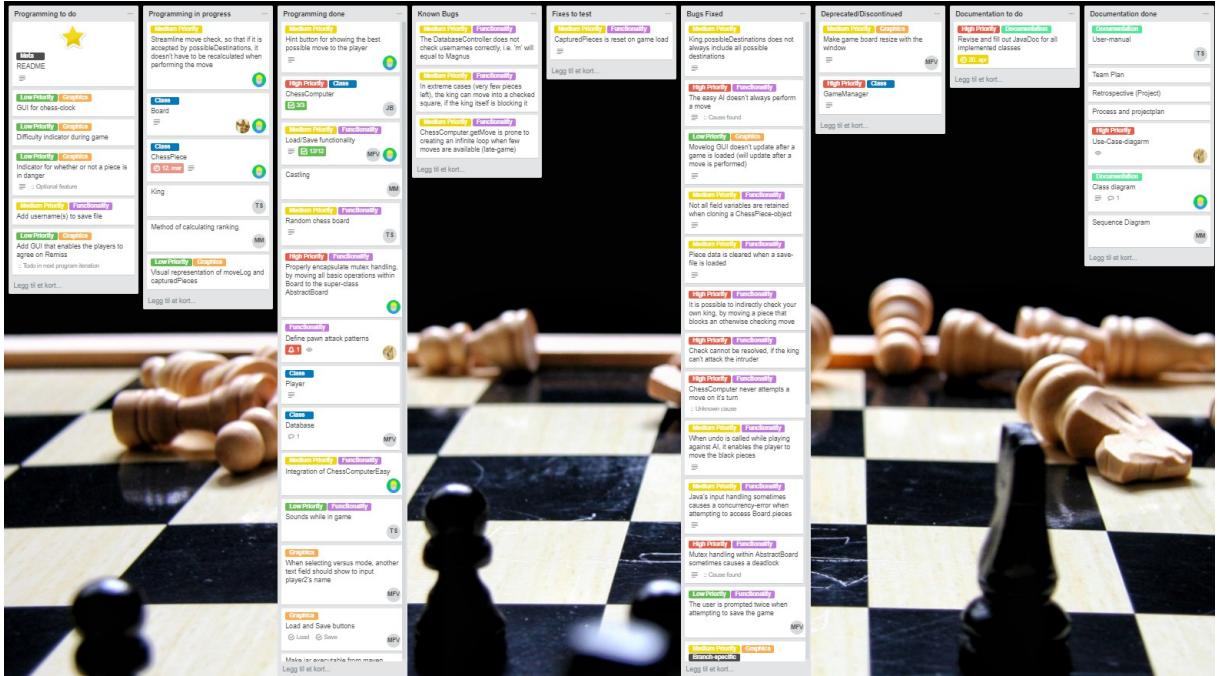


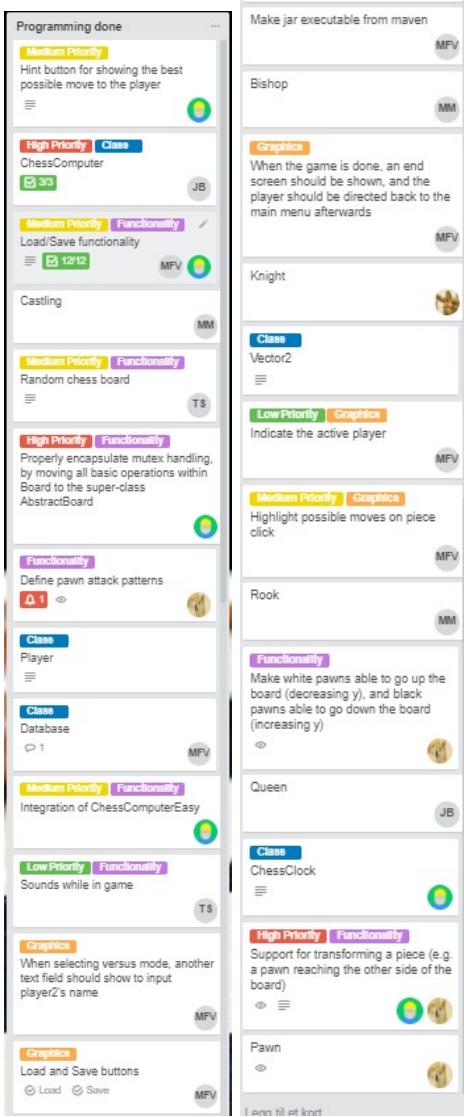


## 4th sprint



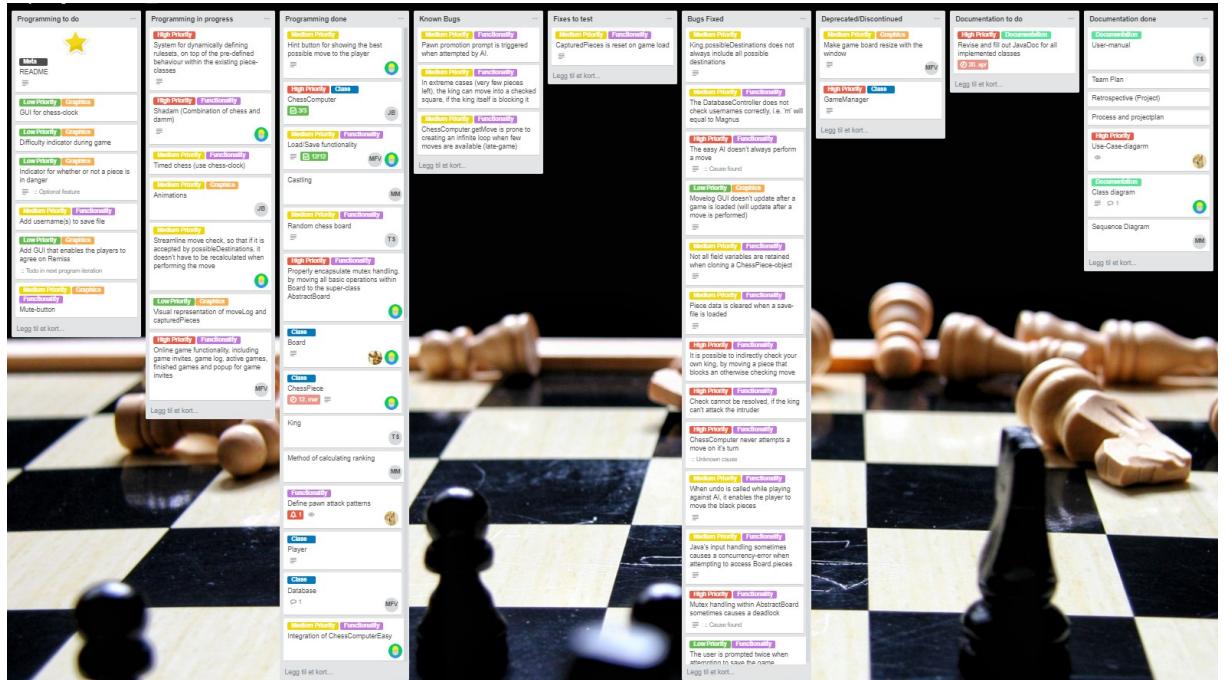






Bugs Fixed	
<b>Medium Priority</b>	King.possibleDestinations does not always include all possible destinations
<b>High Priority</b>	<b>Functionality</b> The easy AI doesn't always perform a move
	:: Cause found
<b>Low Priority</b>	<b>Graphics</b> Movelog GUI doesn't update after a game is loaded (will update after a move is performed)
<b>Medium Priority</b>	<b>Functionality</b> Not all field variables are retained when cloning a ChessPiece-object
<b>Medium Priority</b>	<b>Functionality</b> Piece data is cleared when a save-file is loaded
<b>High Priority</b>	<b>Functionality</b> It is possible to indirectly check your own king, by moving a piece that blocks an otherwise checking move
<b>High Priority</b>	<b>Functionality</b> Check cannot be resolved, if the king can't attack the intruder
<b>High Priority</b>	<b>Functionality</b> ChessComputer never attempts a move on its turn
	:: Unknown cause
<b>Medium Priority</b>	<b>Functionality</b> When undo is called while playing against AI, it enables the player to move the black pieces
<b>Medium Priority</b>	<b>Functionality</b> Java's input handling sometimes causes a concurrency-error when attempting to access Board.pieces
<b>High Priority</b>	<b>Functionality</b> Mutex handling within AbstractBoard sometimes causes a deadlock
	:: Cause found
<b>Low Priority</b>	<b>Functionality</b> The user is prompted twice when attempting to save the game
MFV	
<b>Medium Priority</b>	<b>Graphics</b> <b>Branch-specific</b> Buffered moves causes square highlighting to sometimes not appear (moves can still be performed as usual) :: On branch 'bufferedMoves' only
<b>Medium Priority</b>	<b>Functionality</b> When board.movePiece temporarily releases the mutex, it is a possibility that it can be acquired by another waiting process :: Disproved hypothetical
<b>Medium Priority</b>	<b>Functionality</b> Pawn.getPossibleMoves does not correctly reflect behaviour defined by legalMove
<b>Medium Priority</b>	<b>Functionality</b> Under certain circumstances, pawns will attempt to perform en passant on a piece that is not a pawn
<b>Low Priority</b>	<b>Functionality</b> Attacked pieces are not added to board.inactivePieces upon removal
<b>Medium Priority</b>	<b>Functionality</b> Pawns can only perform a doubleStep as their first move
<b>High Priority</b>	<b>Functionality</b> Sometimes after attempting an illegal move, successful moves will not be registered
<b>Medium Priority</b>	<b>Functionality</b> Only black pawns can attack (diagonal move)
King references within board falls out of sync when cloning the board	

## 5th sprint



**Programming done**

**Medium Priority** Hint button for showing the best possible move to the player

**High Priority Class** ChessComputer  
30 J.B.

**Medium Priority Functionality** Load/Save functionality  
12/12 M.F.V. T.S.

Castling M.M.

**Medium Priority Functionality** Random chess board  
T.S.

**High Priority Functionality** Properly encapsulate mutex handling, by moving all basic operations within Board to the super-class AbstractBoard

**Class** Board

**Class** ChessPiece  
12.mor T.S.

King T.S.

Method of calculating ranking M.M.

**Functionality** Define pawn attack patterns  
A.1

**Class** Player

**Class** Database  
1 M.F.V.

**Medium Priority Functionality** Integration of ChessComputerEasy

**Programming done**

**Low Priority Functionality** Sounds while in game

**Graphics** When selecting versus mode, another text field should show to input player2's name M.F.V.

**Graphics** Load and Save buttons  
Load Save M.F.V.

Make jar executable from maven M.F.V.

Bishop M.M.

**Graphics** When the game is done, an end screen should be shown, and the player should be directed back to the main menu afterwards M.F.V.

Knight M.F.V.

**Class** Vector2

**Low Priority Graphics** Indicate the active player M.F.V.

**Medium Priority Graphics** Highlight possible moves on piece click M.F.V.

Rook M.M.

**Functionality** Make white pawns able to go up the board (decreasing y), and black pawns able to go down the board (increasing y)  
Q.P. J.B.

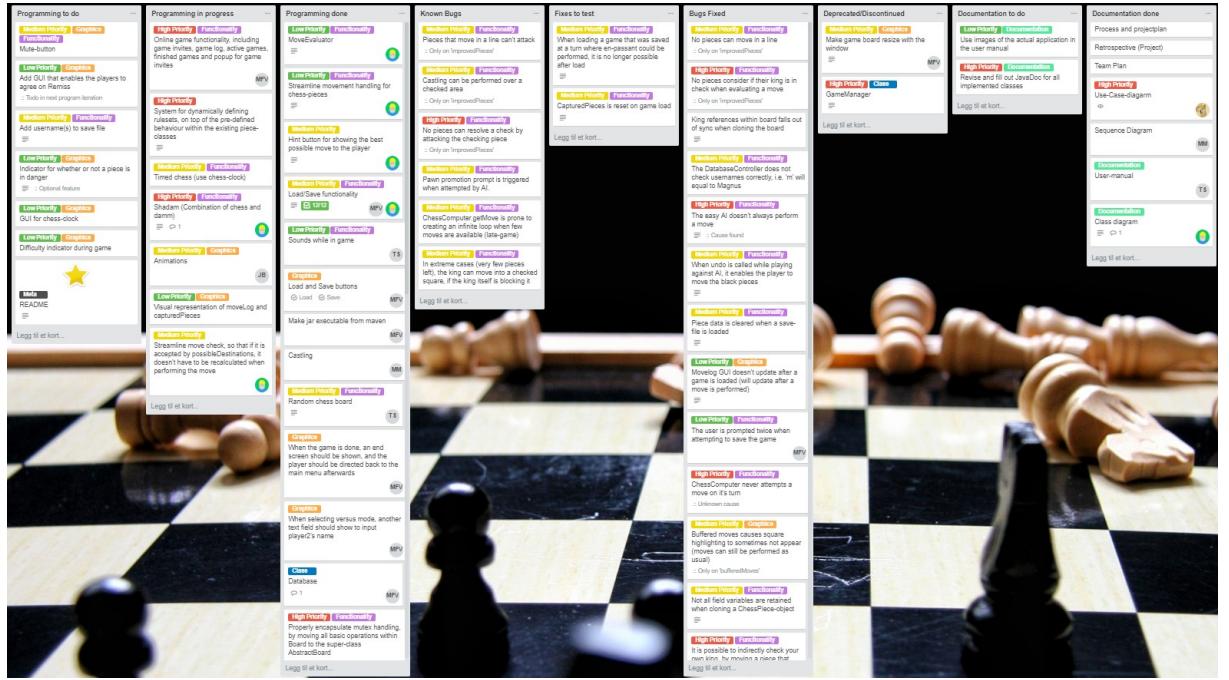
Queen J.B.

**Class** ChessClock

**High Priority Functionality** Support for transforming a piece (e.g. a pawn reaching the other side of the board)

Pawn

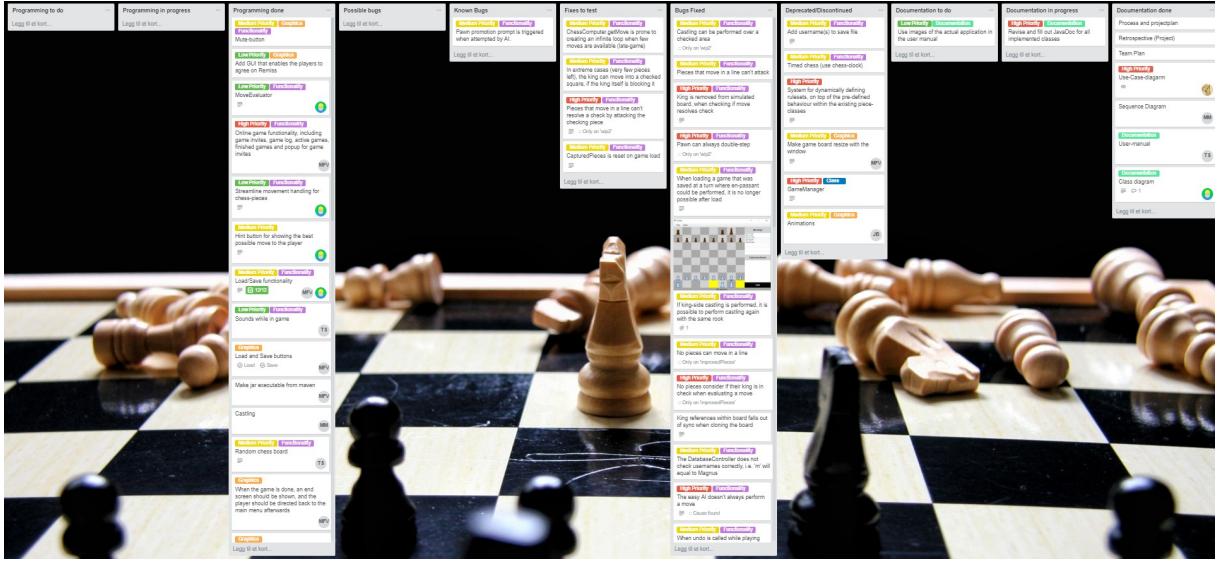
Bugs Fixed	
<b>Medium Priority</b>	King.possibleDestinations does not always include all possible destinations
<b>Medium Priority</b>	The DatabaseController does not check usernames correctly, i.e. 'm' will equal to Magnus
<b>High Priority</b>	The easy AI doesn't always perform a move :: Cause found
<b>Low Priority</b>	Movelog GUI doesn't update after a game is loaded (will update after a move is performed)
<b>Medium Priority</b>	Not all field variables are retained when cloning a ChessPiece-object
<b>Medium Priority</b>	Piece data is cleared when a save-file is loaded
<b>High Priority</b>	It is possible to indirectly check your own king, by moving a piece that blocks an otherwise checking move
<b>High Priority</b>	Check cannot be resolved, if the king can't attack the intruder
<b>High Priority</b>	ChessComputer never attempts a move on its turn :: Unknown cause
<b>Medium Priority</b>	When undo is called while playing against AI, it enables the player to move the black pieces
<b>Medium Priority</b>	Java's input handling sometimes causes a concurrency-error when attempting to access Board.pieces
<b>High Priority</b>	Mutex handling within AbstractBoard sometimes causes a deadlock :: Cause found
<b>Low Priority</b>	The user is prompted twice when attempting to save the game
<b>Medium Priority</b>	Buffered moves causes square highlighting to sometimes not appear (moves can still be performed as usual) :: On branch 'bufferedMoves' only
<b>Medium Priority</b>	When board.movePiece temporarily releases the mutex, it is a possibility that it can be acquired by another waiting process :: Disproved hypothetical
<b>Medium Priority</b>	Pawn.getPossibleMoves does not correctly reflect behaviour defined by legalMove
<b>Medium Priority</b>	Under certain circumstances, pawns will attempt to perform en passant on a piece that is not a pawn
<b>Low Priority</b>	Attacked pieces are not added to board.inactivePieces upon removal
<b>Medium Priority</b>	Pawns can only perform a doubleStep as their first move
<b>High Priority</b>	Sometimes after attempting an illegal move, successful moves will not be registered
<b>Medium Priority</b>	Only black pawns can attack (diagonal move)
<b>High Priority</b>	King references within board falls out of sync when cloning the board
<b>Medium Priority</b>	Pawns can only perform a doubleStep as their first move
<b>Medium Priority</b>	Only black pawns can attack (diagonal move)



**Programming done**

Category	Description	Priority	Status
Low Priority / Functionality	MoveEvaluator	Low Priority	MM
Low Priority / Functionality	Streamline movement handling for chess-pieces	Low Priority	MM
Medium Priority	Hint button for showing the best possible move to the player	Medium Priority	MM
Medium Priority / Functionality	Load/Save functionality	Medium Priority	MFV
Low Priority / Functionality	Sounds while in game	Low Priority	T\$
Graphics	Load and Save buttons	MFV	Load Save
	Make jar executable from maven	MFV	
	Castling	MM	
Medium Priority / Functionality	Random chess board	Medium Priority	T\$
Graphics	When the game is done, an end screen should be shown, and the player should be directed back to the main menu afterwards	MFV	
Graphics	When selecting versus mode, another text field should show to input player2's name	MFV	
Class	Database	MFV	1
High Priority / Functionality	Properly encapsulate mutex handling, by moving all basic operations within Board to the super-class AbstractBoard	High Priority	MM
Programming done	Method of calculating ranking	MM	
Medium Priority / Functionality	Integration of ChessComputerEasy	Medium Priority	MM
Functionality	Make white pawns able to go up the board (decreasing y), and black pawns able to go down the board (increasing y)	MM	
Low Priority / Graphics	Indicate the active player	MFV	
Medium Priority / Graphics	Highlight possible moves on piece click	MFV	
High Priority / Class	ChessComputer	High Priority	JB
	Rook	MM	
High Priority / Functionality	Support for transforming a piece (e.g. a pawn reaching the other side of the board)	High Priority	MM
Functionality	Define pawn attack patterns	MM	1
Class	ChessClock	MM	
Class	Player	MM	
Class	Vector2	MM	
Class	Board	MM	
Class	ChessPiece	MM	12.mer

Bugs Fixed	
<b>Medium Priority</b>	<b>Functionality</b>
No pieces can move in a line :: Only on 'improvedPieces'	
<b>High Priority</b>	<b>Functionality</b>
No pieces consider if their king is in check when evaluating a move :: Only on 'improvedPieces'	
King references within board falls out of sync when cloning the board	
<b>Medium Priority</b>	<b>Functionality</b>
The DatabaseController does not check usernames correctly, i.e. 'm' will equal to Magnus	
<b>High Priority</b>	<b>Functionality</b>
The easy AI doesn't always perform a move :: Cause found	
<b>Medium Priority</b>	<b>Functionality</b>
When undo is called while playing against AI, it enables the player to move the black pieces	
<b>Medium Priority</b>	<b>Functionality</b>
Piece data is cleared when a save-file is loaded	
<b>Low Priority</b>	<b>Graphics</b>
Movelog GUI doesn't update after a game is loaded (will update after a move is performed)	
<b>Low Priority</b>	<b>Functionality</b>
The user is prompted twice when attempting to save the game	
MFV	
<b>High Priority</b>	<b>Functionality</b>
ChessComputer never attempts a move on its turn :: Unknown cause	
<b>Medium Priority</b>	<b>Graphics</b>
Buffered moves causes square highlighting to sometimes not appear (moves can still be performed as usual) :: Only on 'bufferedMoves'	
<b>Medium Priority</b>	<b>Functionality</b>
Not all field variables are retained when cloning a ChessPiece-object	
<b>High Priority</b>	<b>Functionality</b>
It is possible to indirectly check your own king, by moving a piece that blocks an otherwise checking move	
<b>High Priority</b>	<b>Functionality</b>
Mutex handling within AbstractBoard sometimes causes a deadlock :: Cause found	
<b>Medium Priority</b>	<b>Functionality</b>
When board.movePiece temporarily releases the mutex, it is a possibility that it can be acquired by another waiting process :: Disproved hypothetical	
<b>High Priority</b>	<b>Functionality</b>
Check cannot be resolved, if the king can't attack the intruder	
<b>Medium Priority</b>	<b>Functionality</b>
Java's input handling sometimes causes a concurrency-error when attempting to access Board.pieces	
<b>Medium Priority</b>	
King.possibleDestinations does not always include all possible destinations	
<b>Medium Priority</b>	<b>Functionality</b>
Under certain circumstances, pawns will attempt to perform en passant on a piece that is not a pawn	
<b>Low Priority</b>	<b>Functionality</b>
Attacked pieces are not added to board.inactivePieces upon removal	
<b>Medium Priority</b>	<b>Functionality</b>
Pawn.getPossibleMoves does not correctly reflect behaviour defined by legalMove	
<b>High Priority</b>	<b>Functionality</b>
Sometimes after attempting an illegal move, successful moves will not be registered	
<b>Medium Priority</b>	<b>Functionality</b>
Pawns can only perform a doubleStep as their first move	
<b>Medium Priority</b>	<b>Functionality</b>
Only black pawns can attack (diagonal move)	



**Programming done**

- Medium Priority | Graphics**  
Mute-button
- Low Priority | Graphics**  
Add GUI that enables the players to agree on Remiss
- Low Priority | Functionality**  
MoveEvaluator
- High Priority | Functionality**  
Online game functionality, including game invites, game log, active games, finished games and popup for game invites
- Low Priority | Functionality**  
Streamline movement handling for chess-pieces
- Medium Priority**  
Hint button for showing the best possible move to the player
- Medium Priority | Functionality**  
Load/Save functionality
- Low Priority | Functionality**  
Sounds while in game
- Graphics**  
Load and Save buttons
- Make jar executable from maven
- Castling
- Medium Priority | Functionality**  
Random chess board
- Graphics**  
When the game is done, an end screen should be shown, and the player should be directed back to the main menu afterwards

**Graphics**  
When selecting versus mode, another text field should show to input player2's name

**Class**  
Database

**High Priority | Functionality**  
Properly encapsulate mutex handling, by moving all basic operations within Board to the super-class AbstractBoard

Method of calculating ranking

**Integration of ChessComputerEasy**

**Functionality**  
Make white pawns able to go up the board (decreasing y), and black pawns able to go down the board (increasing y)

**Low Priority | Graphics**  
Indicate the active player

**Medium Priority | Graphics**  
Highlight possible moves on piece click

**High Priority | Class**  
ChessComputer  
**3/3**

Rook

**High Priority | Functionality**  
Support for transforming a piece (e.g. a pawn reaching the other side of the board)

**Functionality**  
Define pawn attack patterns

**Queen**

**King**

**Bishop**

**Knight**

**Pawn**

**ChessClock**

**Player**

**Vector2**

**Board**

**ChessPiece**  
**12.mcr**

Bugs Fixed	
<b>Medium Priority</b>	<b>Functionality</b>
Casting can be performed over a checked area :: Only on 'wip2'	
<b>Medium Priority</b>	<b>Functionality</b>
Pieces that move in a line can't attack	
<b>High Priority</b>	<b>Functionality</b>
King is removed from simulated board, when checking if move resolves check	
<b>High Priority</b>	<b>Functionality</b>
Pawn can always double-step :: Only on 'wip2'	
<b>Medium Priority</b>	<b>Functionality</b>
When loading a game that was saved at a turn where en-passant could be performed, it is no longer possible after load	
<b>Medium Priority</b>	<b>Functionality</b>
If king-side castling is performed, it is possible to perform castling again with the same rook ⊕ 1	
<b>Medium Priority</b>	<b>Functionality</b>
No pieces can move in a line :: Only on 'ImprovedPieces'	
<b>High Priority</b>	<b>Functionality</b>
No pieces consider if their king is in check when evaluating a move :: Only on 'ImprovedPieces'	
King references within board falls out of sync when cloning the board	
<b>Medium Priority</b>	<b>Functionality</b>
The DatabaseController does not check usernames correctly, i.e. 'm' will equal to Magnus	
<b>High Priority</b>	<b>Functionality</b>
The easy AI doesn't always perform a move ≡ :: Cause found	
<b>Medium Priority</b>	<b>Functionality</b>
When undo is called while playing against AI, it enables the player to move the black pieces	
<b>Medium Priority</b>	<b>Functionality</b>
Piece data is cleared when a save-file is loaded	
<b>Low Priority</b>	<b>Graphics</b>
Movelog GUI doesn't update after a game is loaded (will update after a move is performed)	
<b>Low Priority</b>	<b>Functionality</b>
The user is prompted twice when attempting to save the game	MFV
<b>High Priority</b>	<b>Functionality</b>
ChessComputer never attempts a move on it's turn :: Unknown cause	
<b>Medium Priority</b>	<b>Graphics</b>
Buffered moves causes square highlighting to sometimes not appear (moves can still be performed as usual) :: Only on 'bufferedMoves'	
<b>Medium Priority</b>	<b>Functionality</b>
Not all field variables are retained when cloning a ChessPiece-object	
<b>High Priority</b>	<b>Functionality</b>
It is possible to indirectly check your own king, by moving a piece that blocks an otherwise checking move	
<b>High Priority</b>	<b>Functionality</b>
Mutex handling within AbstractBoard sometimes causes a deadlock ≡ :: Cause found	
<b>Medium Priority</b>	<b>Functionality</b>
When board.movePiece temporarily releases the mutex, it is a possibility that it can be acquired by another waiting process ≡ :: Disproved hypothetical	
<b>High Priority</b>	<b>Functionality</b>
Check cannot be resolved, if the king can't attack the intruder	
<b>Medium Priority</b>	<b>Functionality</b>
Java's input handling sometimes causes a concurrency-error when attempting to access Board.pieces	
<b>Medium Priority</b>	
King.possibleDestinations does not always include all possible destinations	
<b>Medium Priority</b>	<b>Functionality</b>
Under certain circumstances, pawns will attempt to perform an passant on a piece that is not a pawn	
<b>Low Priority</b>	<b>Functionality</b>
Attacked pieces are not added to board.inactivePieces upon removal	
<b>Medium Priority</b>	<b>Functionality</b>
Pawn.getPossibleMoves does not correctly reflect behaviour defined by legalMove	
<b>High Priority</b>	<b>Functionality</b>
Sometimes after attempting an illegal move, successful moves will not be registered	
<b>Medium Priority</b>	<b>Functionality</b>
Pawns can only perform a doubleStep as their first move	
<b>Medium Priority</b>	<b>Functionality</b>
Only black pawns can attack (diagonal move)	