

P1

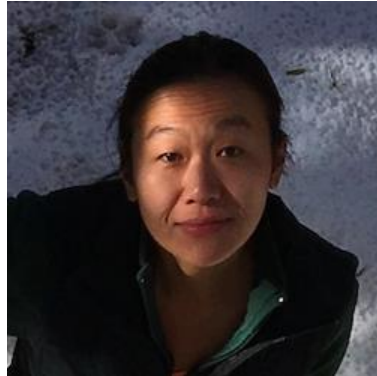
Dedicated RAMs

Axel Wahlstrom, Cathy Lee, Ethan Liem, Hannah VanderHoeven, Najy Faour

Team Members



Hannah VanderHoeven



Cathy Lee

Axel Wahlstrom



Ethan Liem



Najy Faour

Plunder Chess

Summary

- A variation of chess that involves "plundering" the opponent's piece when they are captured.
- Once a player plunders an opponent's piece, their piece is given additional moving capabilities.
- These moving capabilities are known as "vests"
- A vest can only be used for one future move.

User Stories and Tasks

Progress made in P1

Epics



Setup Database

Create the data structure to store users and game states for any game that is currently going on.



Setup Server

Create server side implementation so client can retrieve information about users and games



Login screen

Allow new users to register for an account



Create a match

Create the client and allow users to play games



Play a match

Users can play plunder chess, logic for the game is written



Settings and game status

Create settings, player history, and game statuses

Must Have User Stories

5. 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 move
- The game should tell you what legal moves you can make

- **Acceptance Criteria**

- Pieces only move according to the rules for that piece
- Illegal move for a piece shouldn't change the state of the game
- Pieces can only be moved on the users turn

- **Tasks**

- [48.](#) UI for game board
- [49.](#) UI for pieces
- [50.](#) Implement game states - like turn order, pieces on board, time elapsed
- [51.](#) Server implementation of saving game state
- [52.](#) Game logic for pieces
- [53.](#) Game logic for valid movement
- [68.](#) Implement Plundering
- [76.](#) UI dialog for stealing a piece and using a stolen move

Must Have User Stories

6. As a user, I can delete my account

- **Notes**

- Client should confirm that a user is about to delete account and what that means

- **Acceptance Criteria**

- Deleting an account deletes user information from server
 - Username and password for a deleted account are invalid for logging in

- **Tasks**

- [63.](#) Delete account UI in user setting
 - [64.](#) Implement delete account
 - [65.](#) Send user back to game client register screen

Kanban Board

Product Decisions/Progress made in P1

<> Code ! Issues 71 🔗 Pull requests 1 **Z ZenHub** ⚙ Actions 📁 Projects 📖 Wiki 🛡 Security 📈 Insights

anban ☆ ⇌

Board ⚙

Reports ▾

Roadmap

Workflows **New**

Create...

Edit Workspace

Invite

View Tutorials

Shortcuts

Open in web app

Support and training

💡 Switch to related Workspaces if any repos belong to other Workspaces, allowing for quick access to similar sets of Issues. [See related Workspaces](#) ✕

Repos (1/1) ▾ 🔍 Labels ▾ 📅 Milestones ▾ 👤 Assignees ▾ 📅 Epics ▾ 📄 Releases ▾ ⌚ Estimates ▾ ✍ Authors ▾ 🔍 Find Issues (f+i) **New Issue** +

53 Issues - 0 Story Points ⌵ ⌶
New Issues ⓘ ⚙

cs414-f20-Dedicated... #23
Pick unit testing tool

cs414-f20-Dedicated... #24
Pick continuous integration tool

cs414-f20-Dedicated... #25
Pick a continuous code quality tool

cs414-f20-Dedicated... #27
Add Class Diagram to GitHub

cs414-f20-Dedicated... #28
Finish Creating Tasks from user stories

cs414-f20-DedicatedR... #31
Pick unit testing tool

0 Issues - 0 Story Points ⌵ ⌶
Icebox ⓘ ⚙

Icebox

Low priority, un-prioritized Issues that are not immediate priorities.

16 Issues - 4 Story Points ⌵ ⌶
Backlog ⓘ ⚙

cs414-f20-DedicatedRA... #1
Login Screen
📅 P1
[Filter by Epic Issues](#)

Epic

cs414-f20-Dedicated... #22
Create main page
📅 Login Screen

cs414-f20-DedicatedR... #7
Create window
📅 P1
📅 Login Screen

2 Issues - 0 Story Points ⌵ ⌶
In Progress ⓘ ⚙

cs414-f20-DedicatedR... #19
Update Tasks and Stories

cs414-f20-Dedicated... #26
Pick UI framework
📅 Login Screen

1 Issue - 2 Story Points ⌵ ⌶
Review/Ready to Merge ⓘ ⚙

cs414-f20-DedicatedR... #6
Setup basic Socket and connect to client

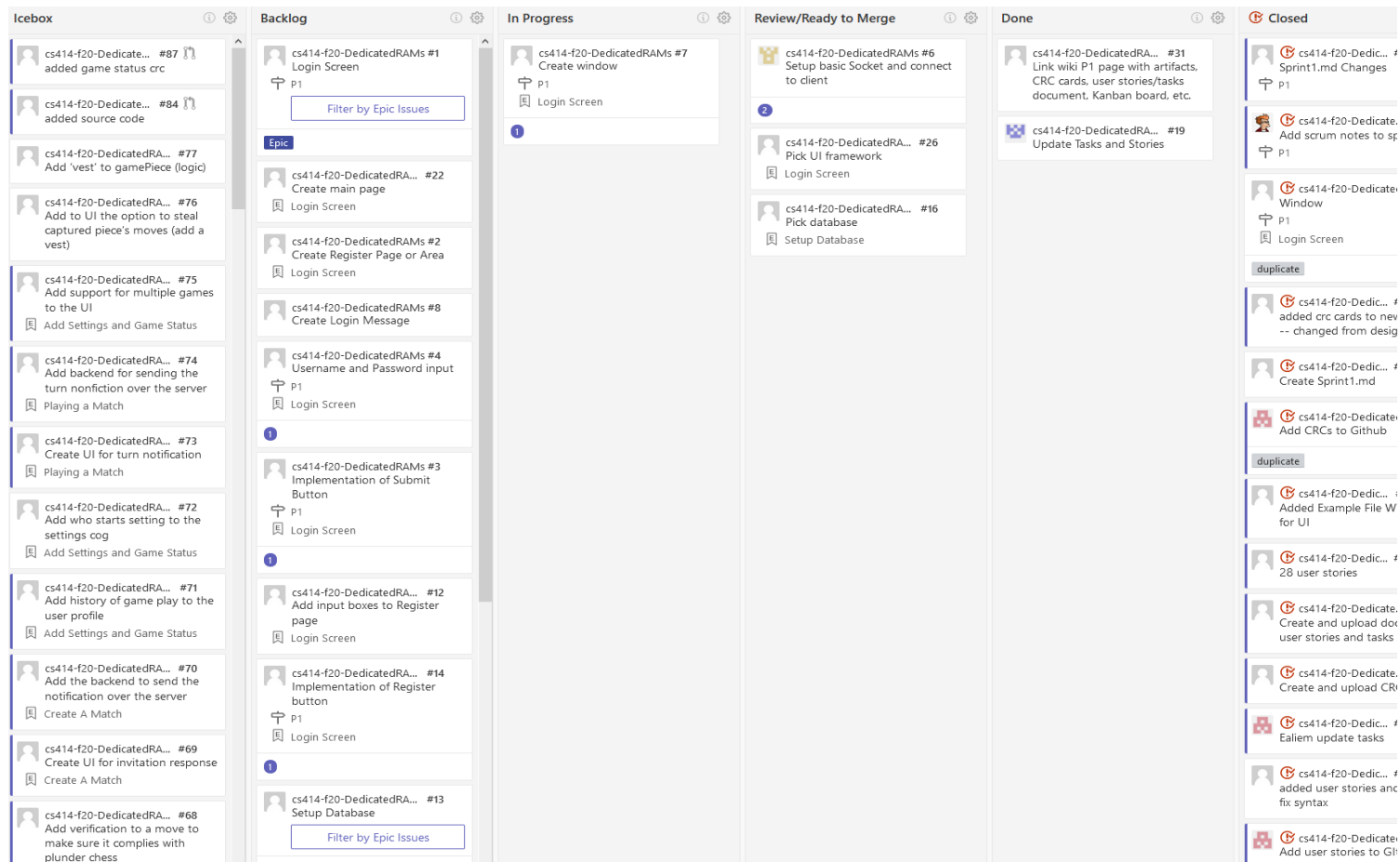
2

0 Issues - 0 Story Points ⌵ ⌶
Done ⓘ ⚙

Issues that be dep

ZenHub – Chrome Extension for Github

Final Sprint Progress



- Finishing up deliverables for P1
- Working on presentation, PowerPoint, and script

Design Artifacts

Progress made in P1

CRC Cards

Game

| Game | |
|---|---|
| Responsibilities | Collaborators |
| <ul style="list-style-type: none">- Has two players- Has a game status- Has a chessboard- Determines who is the winner | <ul style="list-style-type: none">- User(s)- Game Status- Chessboard- Game History- Profile/Account |

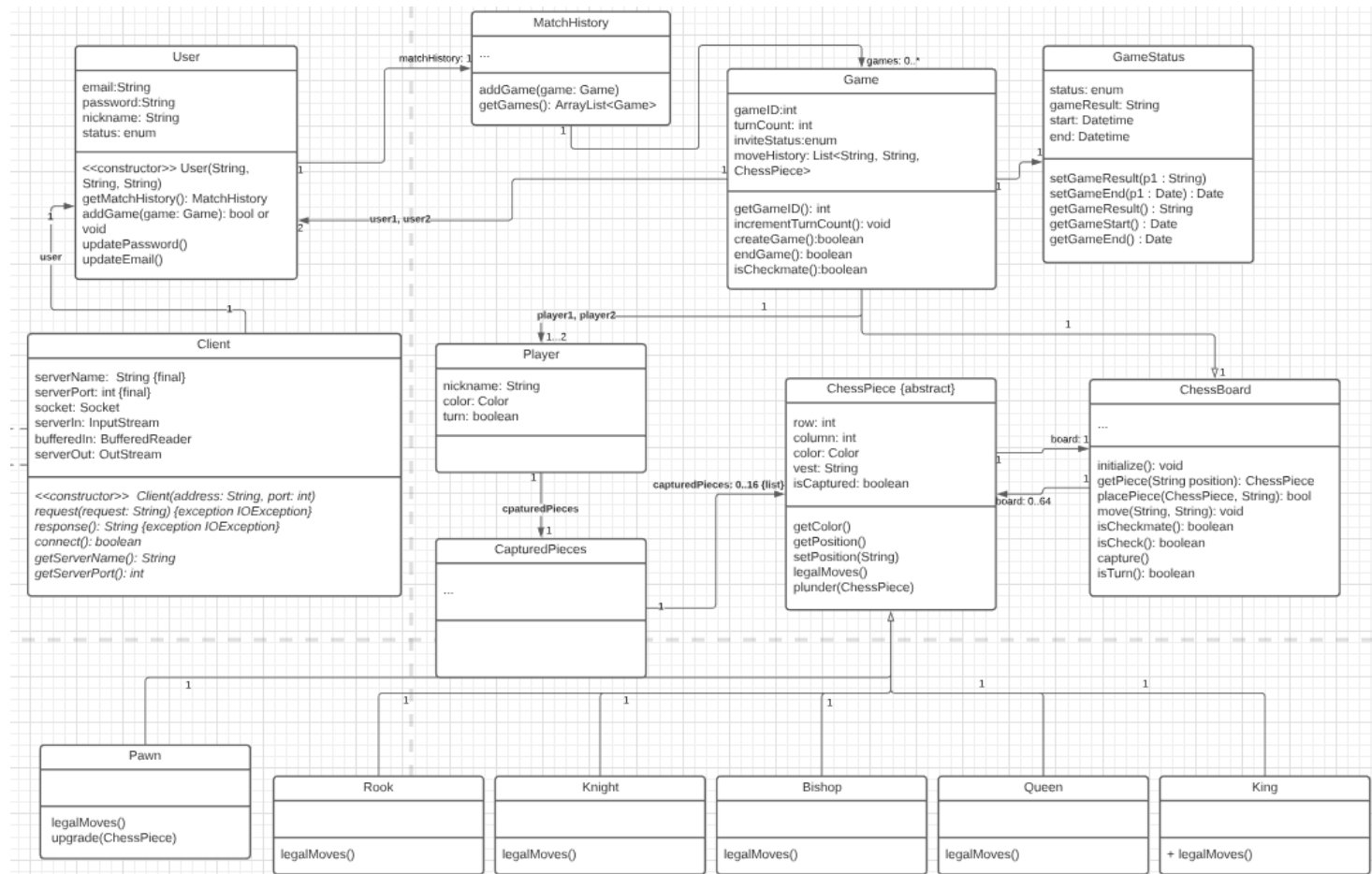
Chess Board

| ChessBoard | |
|---|--|
| Responsibilities | Collaborators |
| <ul style="list-style-type: none">- Moves pieces- Knows when a player is in check/checkmate- Has a turn order | <ul style="list-style-type: none">- Game- Player- ChessPiece |

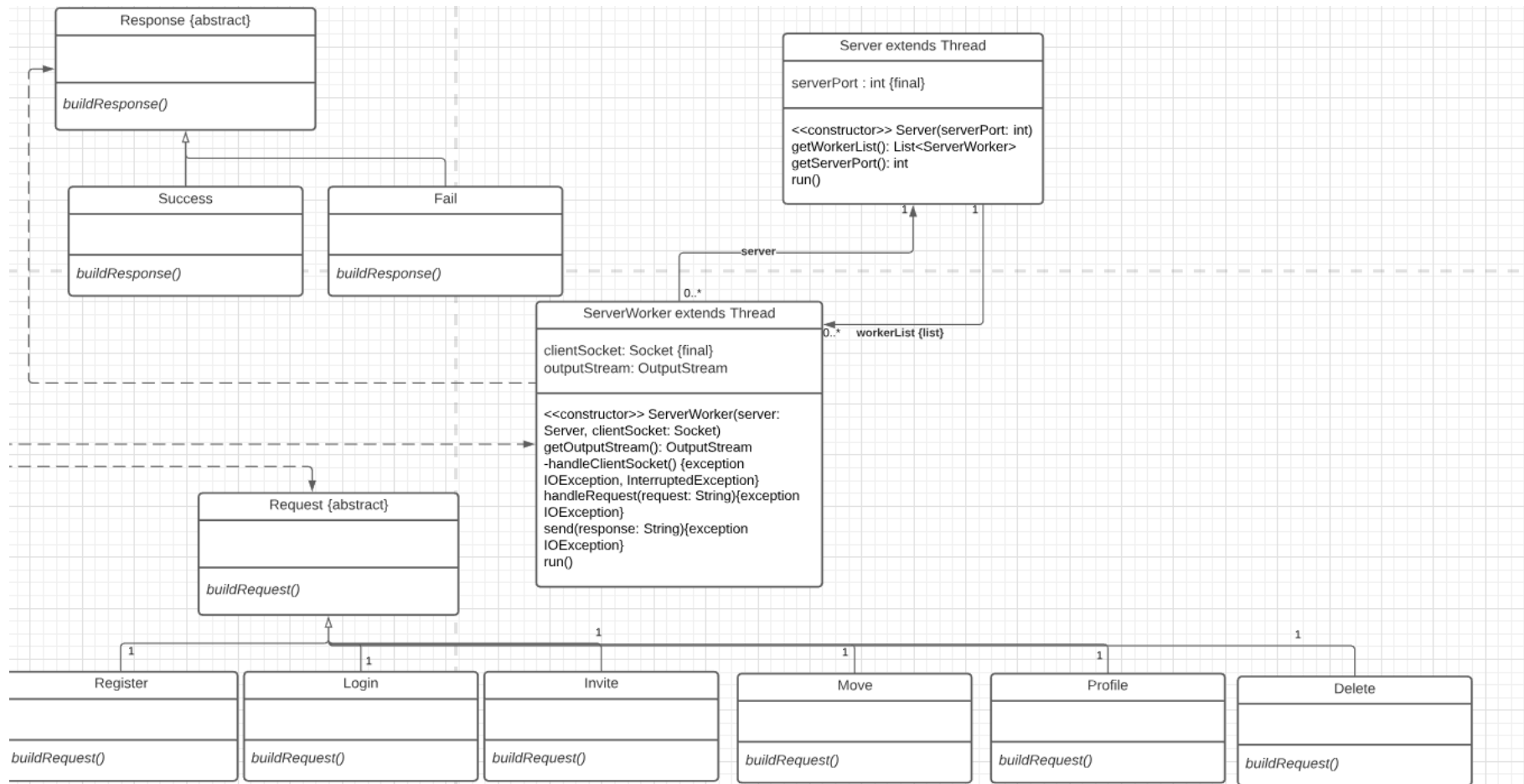
Chess Piece

| ChessPiece | |
|---|---|
| Responsibilities | Collaborators |
| <ul style="list-style-type: none">- Knows its position- Has a color- Plunders another piece's moves- Belongs to a Chessboard- Is either a rook, bishop, knight, pawn, queen, king- Knows its legal moves depending on what piece it is | <ul style="list-style-type: none">- Player- Chessboard |

UML Diagram



Part 1



Part 2

Output of Scrum Ceremonies

Sprint #1 - Dedicated RAMs

Goal

- Prepare User Stories and Design Artifacts to allow organized, intentional development in future Sprints.
- **Scrum Master:** *Axel Wahlstrom*
- **Policies**
 - GitHub etiquette
- **Plan**
 - Epics planned for this release.
 - N/A

Definition of Done

- User Stories and tasks uploaded to GitHub and linked with a Kanban board.
- Class-Responsibility-Collaboration (CRC) Cards completed (to their fullest extent) and added to GitHub.
- Kanban board screenshots uploaded to GitHub populated with project tasks and epics.
- Class Diagrams completed (to its fullest extent) and uploaded to GitHub.
- Scrum Ceremonies (sprint review and retro) uploaded to GitHub
- Deliverables linked in the GitHub wiki for the team's repository.
- Presentation (video) of deliverables at the end of the sprint.
- Individual peer-evaluations at the end of the sprint.

Scrum Meetings

Sprint #1 - Dedicated RAMs

Faour, Najy

| Date | Tasks in progress | Impediments |
|----------|---|-------------|
| 09-11-20 | Looking into Server/Client communication code | |
| 09-14-20 | Editing project deliverables | |
| 09-16-20 | Server-Client code | |
| 09-18-20 | Server Protocol & Schemas as well as CRC and Class Diagrams | |
| 09-21-20 | Server Protocol | |
| 09-23-20 | Scrum team going over deliverables before presentation/Server/Client code | |

Source Code

Product Decisions

Source Code Summery

- Java Front End
 - Java Swing
- Database
 - My SQL
- Started Basic Client/Server Interaction

Review

- Completed Tasks:
 - CRC Cards
 - Class Diagram
 - User Stories and Acceptance Criteria
 - Daily standup output
 - P1 Dedicated RAMs Wiki page
- What went well:
 - Tasks from this sprint were completed in a timely matter and everyone worked effectively together to get the sprint finished by the cutoff date.

Retrospective

- Problems Encountered:
 - Avoid using words such as 'can' or 'lets' when designing CRC cards
 - More descriptive wording on CRC cards
 - Line notation for class Diagram
 - Resolution: enforce stricter policies when designing artifacts
- What went well:
 - Communication
 - Good teamwork to get tasks done