

# P2 Jungle App

From `Team.name()`

Alex Bailey  
David Wells  
Tim Rooney  
Brian Crane

# Scrum Ceremony Outputs

How we Improved:

- Increased communication
- More frequent Scrum Meetings
- Pair programming
- Followed the lead of the Scrum Master
- Met with Product Owner multiple times to make sure our development and user stories were in line with what was expected
- Did Code Refactoring to improve scalability and ease of development

# P2 Sprint Retrospective

## What went well:

- Met with product owner and prioritize user stories
- Refactored previous user stories and tasks to include Acceptance Criteria
- Focused on completing the deliverables as well as completing tasks
- Communicated between teammates more than previous sprint (via Slack, Scrum, Planning Meetings)
- Participated in more pair programming
- Took a more explicit focus about following scrum methodology
- Took inspiration from good code practices from class and applied it to code

# P2 Sprint Retrospective

## Room for improvement

- Continue to redistribute workload evenly between remaining members
- Did work before planning partly due to not having feedback
- Worked in spurts rather than a constant workflow
- Greater test coverage, especially on the Front-End

# P2 Sprint Retrospective

## Future Plans:

- For next sprint we plan to do extensive Design and Planning Meetings before starting to code features
- Continue good communication, pair programming, and frequent meetups
- Continue meeting with product owner to prioritize stories efficiently
- Aim for a constant work flow

# User stories and tasks

## Highest Priority User Stories:

- As a user, I would like to be able to create and play games
- As a user, I would like to know when no moves are possible
- As a user, I would like to create/delete accounts and login/logout
- As a user, I would like to be able to send and accept invites

# User stories and tasks

Examples of tasks that were completed in order to fulfill these Stories:

- “As a user I want to be able to look at the board, and see traps, dens, pieces, and available moves. I want all the information I would need to make an educated move.”
- “Implement move validity detection for determining which squares a selected piece may move to, so that highlighted tiles for the client may be updated.”
- “As a user, I would like to be able to log in and log out of the game through the client application interface.”
- “When a user accepts an invite, the game should be created.”

# Kanban Board Developments

Repos (1/1) ▾

Labels ▾

Milestones ▾

Assignees ▾

Epics ▾

Releases ▾

Estimates ▾

Authors ▾

Find Issues (f+i)

New Issue +

35 Issues - 69 Story Points

New Issues

cs414-f18-001-Team.n... #104

GameServer handle create invite

Project 2

As a user, I would like to be able...

enhancement

cs414-f18-001-Team.n... #108

Database invite support

Project 2

As a user, I would like to be able...

enhancement

cs414-f18-001-Team.n... #109

Client support of invite functionality

Project 2

As a user, I would like to be able...

enhancement

cs414-f18-001-Team.n... #113

Database support of game creation

Project 2

As a user, I would like to be able...

enhancement

cs414-f18-001-Team.n... #114

Database support of game piece moves

Project 2

As a user, I would like to be able...

enhancement

cs414-f18-001-Team.n... #115

GameServer view match history

Project 2

As a user, I would like to be able...

enhancement

3 Issues - 0 Story Points

Icebox

cs414-f18-001-Team.na... #72

As a user, I would like to be able to chat with other players (6)

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #74

As a user, I would like to be able to play in tournaments (6)

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.n... #158

As a user, I would like to play the game vs an AI (6)

Project 2

Filter by Epic Issues

Epic

3 Issues - 2 Story Points

Backlog

cs414-f18-001-Team.n... #120

GameServer unregister user API support

Project 2

As a user, I would like to create/...

enhancement

cs414-f18-001-Team.na... #13

As a user, I would like to be able to view match histories (5)

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.n... #139

Add MariaDB interaction functionality to check DB table state after each API is invoked

Project 2

As a developer, it is important th...

test

7 Issues - 13 Story Points

In Progress

cs414-f18-001-Team.na... #12

As a user, I would like to create/delete and login/logout of my user account (2)

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #70

As a user, I would like to be able to send and accept invites (3)

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #35

As a developer, it is important that we test our code

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #77

Set up Selenium Framework for Front End

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #33

As a developer client-side code should be tested

Project 2

Filter by Epic Issues

Epic

2 Issues - 2 Story Points

Review/QA

cs414-f18-001-Team.na... #99

Add images to game

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #149

Tprooney images

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #162

Add Kanban Board screenshot for 29th of October

Project 2

Filter by Epic Issues

Epic

1 Issue - 0 Story Points

Done

cs414-f18-001-Team.na... #128

Create methods to invoke GET and POST on RESTful server

Project 2

Filter by Epic Issues

Epic

48+ Issues - 48 Story Points

Closed

cs414-f18-001-Team.na... #73

As a user, I would like to play the game vs an AI (6)

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #71

As a user, I would like to know when no moves are possible (1)

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #150

renamed den image, made conditions to handle null images, minimal css...

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #148

David wells the developer patch 1

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #147

Lucidchart diagrams should not be links

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #146

Added class diagram and CRC cards for gameLogic

Project 2

Filter by Epic Issues

Epic

cs414-f18-001-Team.na... #144

Deliverables folders

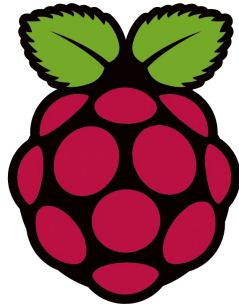
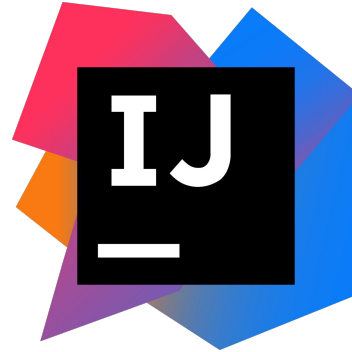
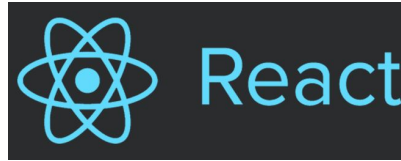
Project 2

Filter by Epic Issues

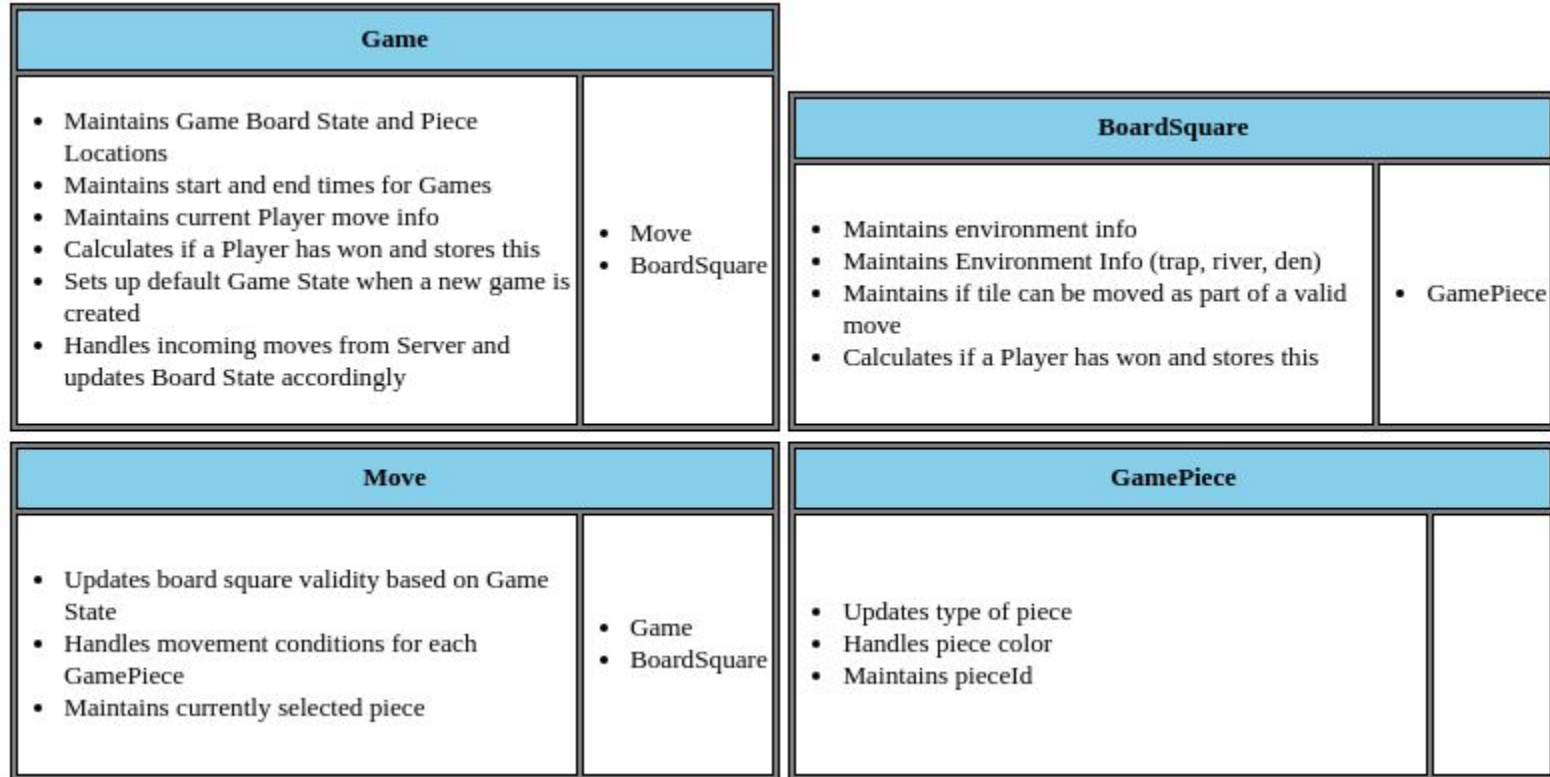
Epic



# What We're Using



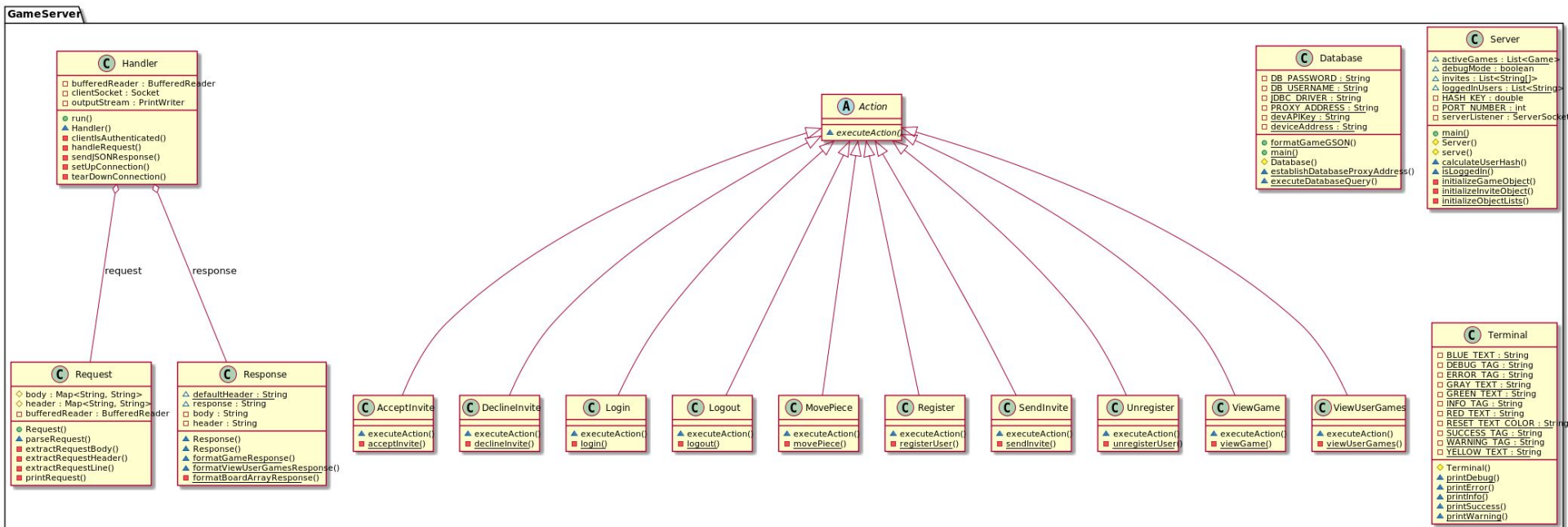
# GameLogic CRC



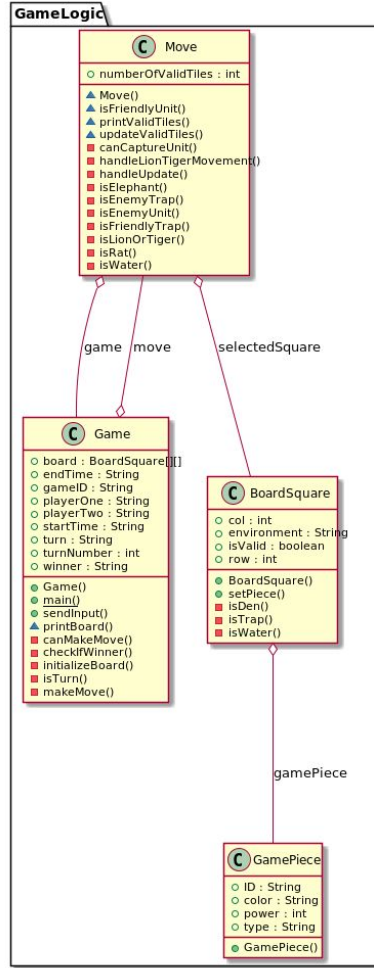
# Server Logic CRC

<b>AcceptInvite</b> <ul style="list-style-type: none"> <li>Support the acceptance of a game invite</li> </ul> <ul style="list-style-type: none"> <li>Database</li> <li>Terminal</li> <li>Server</li> <li>Request</li> <li>Response</li> </ul>	<b>Action</b> <ul style="list-style-type: none"> <li>Abstract class for all concrete executors of client requests</li> </ul> <ul style="list-style-type: none"> <li>Request</li> <li>Response</li> </ul>	<b>Database</b> <ul style="list-style-type: none"> <li>Handle all queries to the database</li> <li>Handle connections to the database</li> <li>Format data for storing in the database</li> <li>Allow users to execute custom queries from the terminal</li> </ul> <ul style="list-style-type: none"> <li>Terminal</li> <li>Game</li> </ul>	<b>DeclineInvite</b> <ul style="list-style-type: none"> <li>Support the decline of a game invite</li> </ul> <ul style="list-style-type: none"> <li>Database</li> <li>Server</li> <li>Terminal</li> <li>Request</li> <li>Response</li> </ul>
<b>Handler</b> <ul style="list-style-type: none"> <li>Handles set up and tear down of connections with clients</li> <li>Invokes Action classes</li> <li>Sends responses to clients</li> </ul> <ul style="list-style-type: none"> <li>Request</li> <li>Response</li> <li>Terminal</li> <li>Action classes</li> </ul>	<b>Login</b> <ul style="list-style-type: none"> <li>Support the login of a client</li> </ul> <ul style="list-style-type: none"> <li>Terminal</li> <li>Server</li> <li>Request</li> <li>Response</li> <li>Database</li> </ul>	<b>Logout</b> <ul style="list-style-type: none"> <li>Support the logout of a client</li> </ul> <ul style="list-style-type: none"> <li>Terminal</li> <li>Server</li> <li>Request</li> <li>Response</li> </ul>	<b>MovePiece</b> <ul style="list-style-type: none"> <li>Support the moving of a piece</li> </ul> <ul style="list-style-type: none"> <li>Terminal</li> <li>Server</li> <li>Request</li> <li>Response</li> <li>Database</li> </ul>
<b>Register</b> <ul style="list-style-type: none"> <li>Support the registration of a new user</li> </ul> <ul style="list-style-type: none"> <li>Terminal</li> <li>Server</li> <li>Request</li> <li>Response</li> <li>Database</li> </ul>	<b>Request</b> <ul style="list-style-type: none"> <li>Extract and store user request data</li> </ul> <ul style="list-style-type: none"> <li>Terminal</li> </ul>	<b>Response</b> <ul style="list-style-type: none"> <li>Format and store the server response to the client</li> </ul> <ul style="list-style-type: none"> <li>Game</li> <li>BoardSquare</li> <li>Terminal</li> </ul>	<b>SendInvite</b> <ul style="list-style-type: none"> <li>Support the sending of a game invite</li> </ul> <ul style="list-style-type: none"> <li>Terminal</li> <li>Server</li> <li>Request</li> <li>Response</li> <li>Database</li> </ul>
<b>Server</b> <ul style="list-style-type: none"> <li>Initiate the server process</li> <li>Initialize the state lists</li> <li>Accept connections from clients, spawn Handlers</li> </ul> <ul style="list-style-type: none"> <li>Terminal</li> <li>Game</li> <li>Handler</li> <li>Database</li> </ul>	<b>Terminal</b> <ul style="list-style-type: none"> <li>Provide an interface for classifying and formatting print statements that are useful to the programmer</li> </ul>	<b>Unregister</b> <ul style="list-style-type: none"> <li>Support the unregistration of a user</li> </ul> <ul style="list-style-type: none"> <li>Terminal</li> <li>Request</li> <li>Response</li> <li>Database</li> </ul>	<b>ViewGame</b> <ul style="list-style-type: none"> <li>Support the viewing of a single game</li> </ul> <ul style="list-style-type: none"> <li>Terminal</li> <li>Request</li> <li>Response</li> </ul>
<b>ViewUserGames</b> <ul style="list-style-type: none"> <li>Support the viewing of all games associated with a user</li> </ul> <ul style="list-style-type: none"> <li>Terminal</li> <li>Request</li> <li>Response</li> </ul>			

# Server UML Diagram



# GameLogic UML Diagram



# Changes in our Source Code

Fundamental changes in our product this sprint:

- Added images to our game board
- Increased Login / Logout / Register functionality on both front and back-end
- New JSON format to communicate Board State
- New AvailableMoves being displayed on our board
- New RESTful JUnit tests that automate testing of each API offered by Server
- Refactored Server logic to remove tedious switch statements
- Refactored NodeJS class structure

# Front End Interface

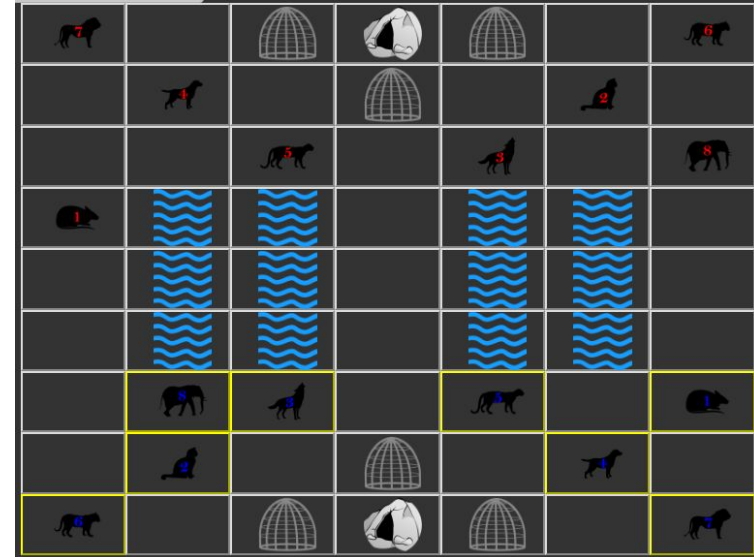
Home Games Game Rules History **Invite** User

Username of player to invite:

dummy\_user

**Login** Register

Username:  Password:



# Changes in our Development Manual this Sprint

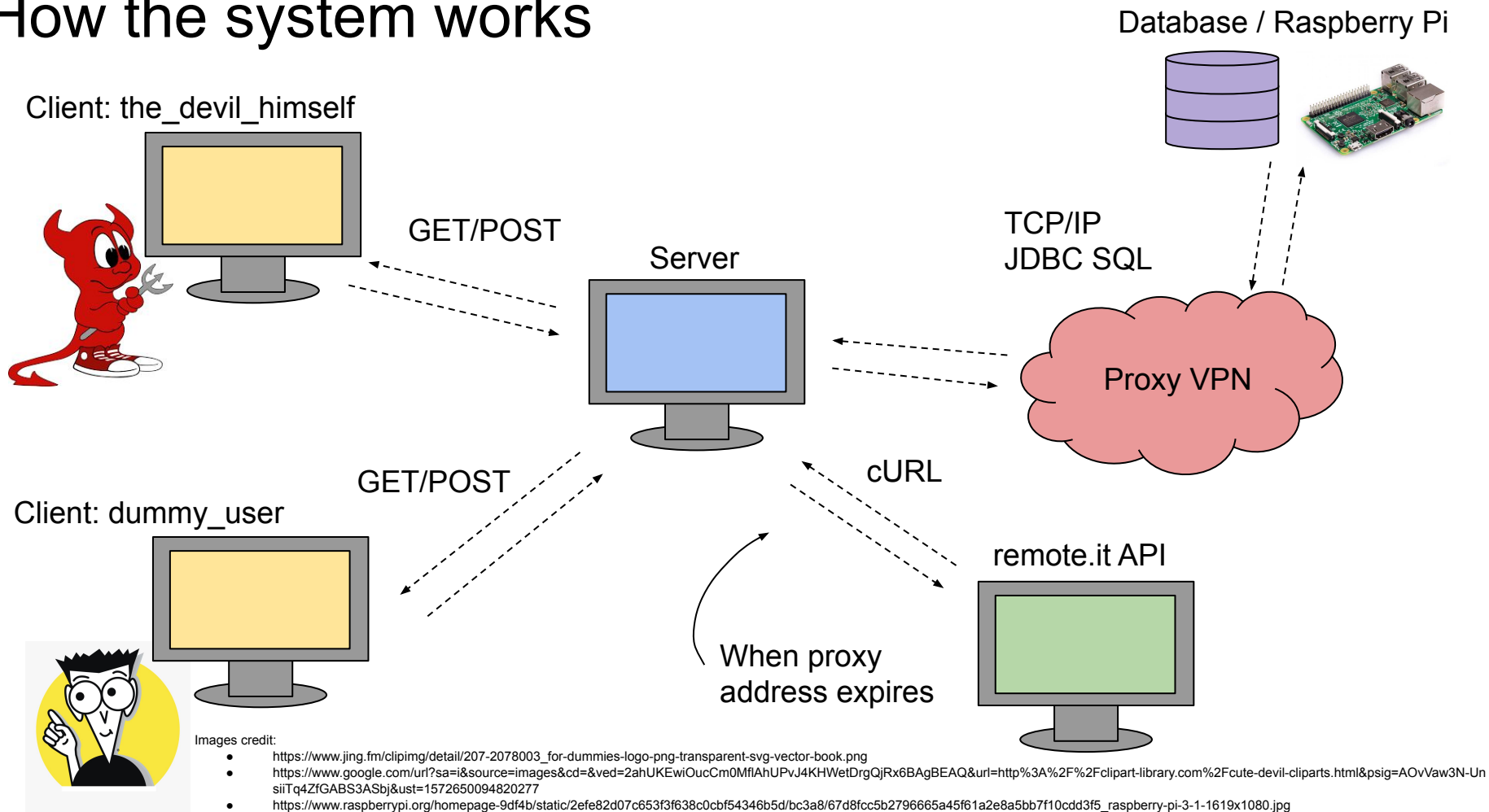
- Currently contains
  - How to set up the work environment
  - Instructions on how to run the client
  - Instructions and commands on how to set up the server
  - Instructions on how to run tests



# Traceability Matrix for this Sprint

[illegible]

# How the system works



# Tech Demo

**Questions?**