Criterion B: Design

Commands list

This list of commands was devised based on the success criteria that the client suggested

Command Name	Description	Arguments	Output
endmatch	Ends current match		Edit message
endvirtual	Ends current virtual solve		Send message
virtual	Starts a virtual solve	Scramble_type	Send message
movelist	show a list of possible moves (cube notation)		Send message
match	1v1 another user	User, scramble_type	Buttons
move	makes a move in a virtual solve	Moves	Send message
addtime	adds a time		Send message
scramble	Generates a nxn scramble	n (size of cube)	Send message
list	Show list of all solves		Send message
stats	Show stats of past solves		Send message
delete	Delete a solve	Solve_id	Send message

Figure 1: First version of commands list

The previous command list was revised after consulting the client (see appendix), leading to the following list

Command Name	Description	Arguments	Output
Endmatch	Ends current match		Edit message
Endvirtual	Ends current virtual solve		Send message

Virtual	Starts a virtual solve	Scramble_type	Send message
Angle	changes angle of camera to x°, y°, z°	x, y, z	Send message
Movelist	show a list of possible moves (cube notation)		Send message
Match	1v1 another user	User, scramble_type	Buttons
Move	makes a move in a virtual solve	Moves	Send message
addtime	adds a time		Modal
3x3	Generates a 3x3 scramble		Send message
2x2	Generates a 2x2 scramble		Send message
4x4	Generates a 4x4 scramble		Send message
5x5	Generates a 5x5 scramble		Send message
6x6	Generates a 6x6 scramble		Send message
7x7	Generates a 7x7 scramble		Send message
list	Show list of all solves		Send message
stats	Show stats of past solves		Send message
details	Shows details of a solve	Solve_id	Send message
Clear	Clears all solves		Send message
delete	Delete a solve	Solve_id	Send message
Ao5	Show the current average of 5		Send message

Ao12	Show the current average of 12		Send message
Mo3	Show the current mean of 3		Send message
bestao5	Show the best average of 5		Send message
bestao12	Show the best average of 12		Send message
bestmo3	Show the best mean of 3		Send message

Figure 2: Revised commands list after consulting client

Figure 3 is the modal (in Discord, modal is a pop-up window with multiple textboxes) that shows after the user uses /addtime. The figure shows the textboxes I chose to add to the modal.

Figure 3: Modal for adding time

Figure 4 and Figure 5 is the embed (in Discord, embed is a feature that allows bots to format and style the messages sent) that shows after the user uses /match and /list.
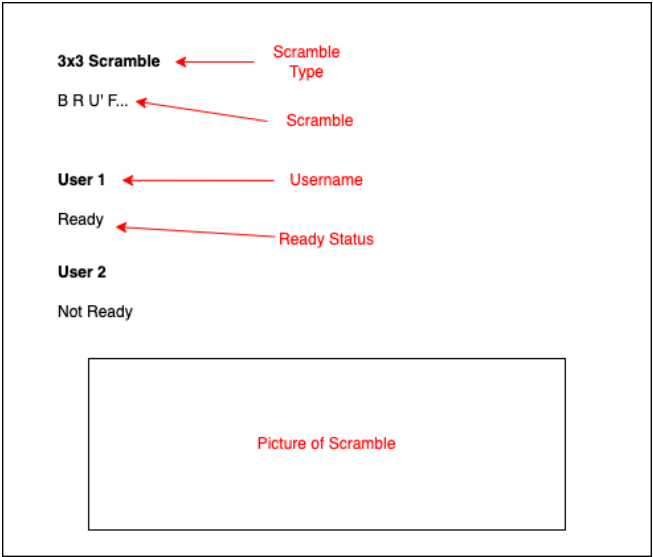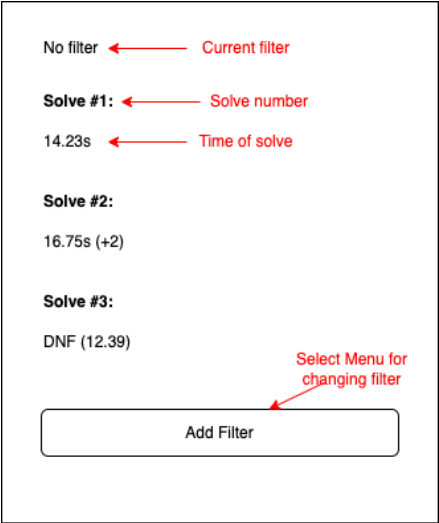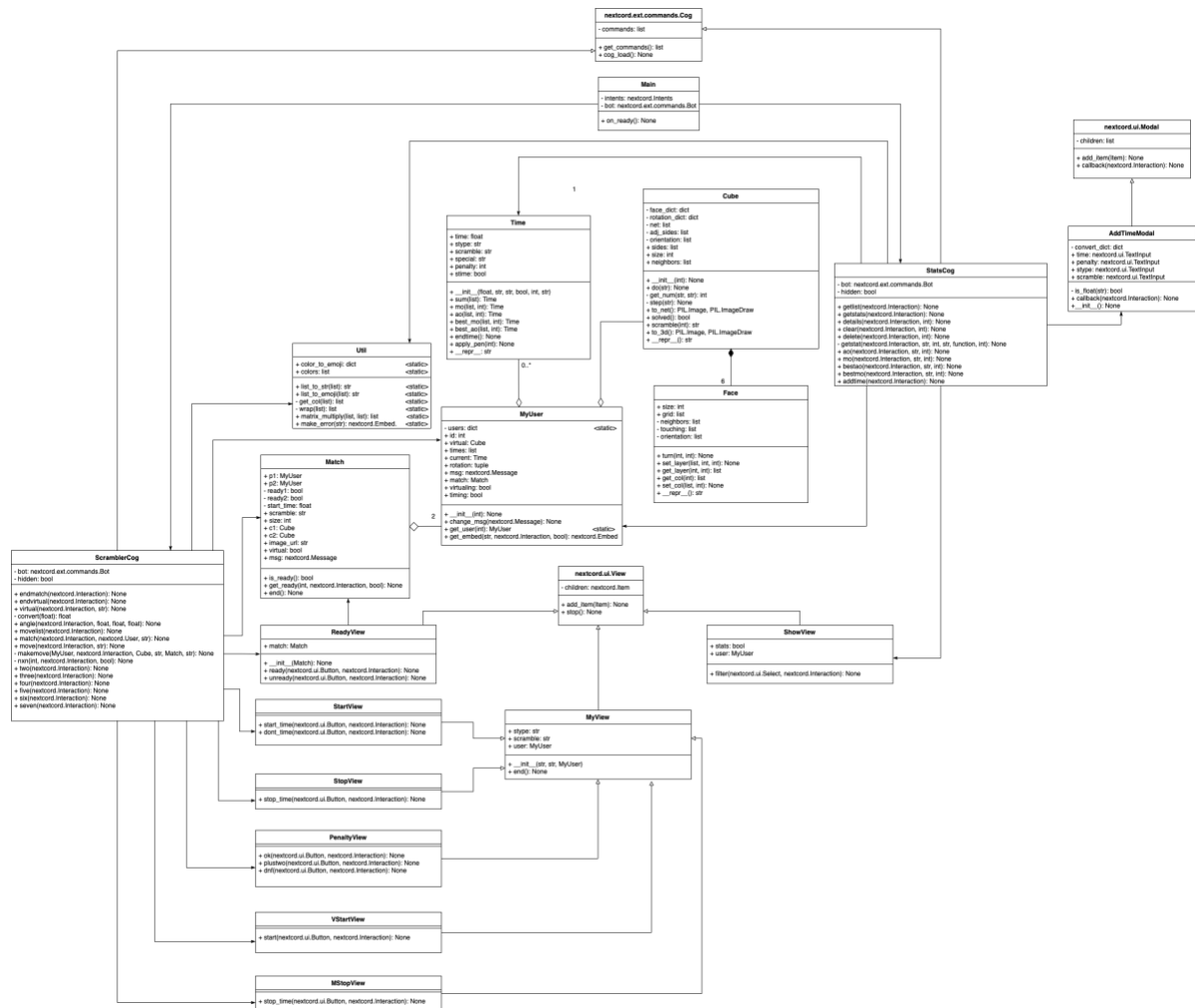
3x3 Scramble ⟵ Scramble
Type

B R U' F... ⟵ Scramble

User 1 ⟵ Username

Ready ⟵ Ready Status

User 2

Not Ready

Picture of Scramble

Figure 4: Embed Layout for /list

No filter ⟵ Current filter

Solve #1: ⟵ Solve number

14.23s ⟵ Time of solve

Solve #2:

16.75s (+2)

Solve #3:

DNF (12.39)

Select Menu for
changing filter

Add Filter

Figure 5: Embed Layout for command /match

Figure 6 is a UML diagram that shows how each class relates to each other. Cogs (StatsCog and ScrambleCog) are classes that organizes a group of commands into a category. Cogs also allow the code to be ran in modules, so if I make changes to the StatsCog, I only have to reload the StatsCog instead of the entire code. These cogs uses the Cube, Time, MyUser, and Match classes, which represents a Rubik's cube, the time and data of a solve, a user, and a 1v1 match between two users, respectively.

Figure 6: UML Diagram

Flowcharts (user experience)

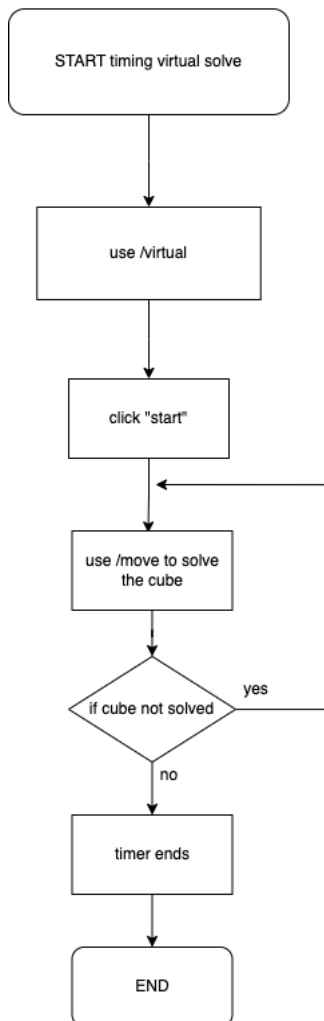The following flowcharts show how the user interacts with the main features of the bot.
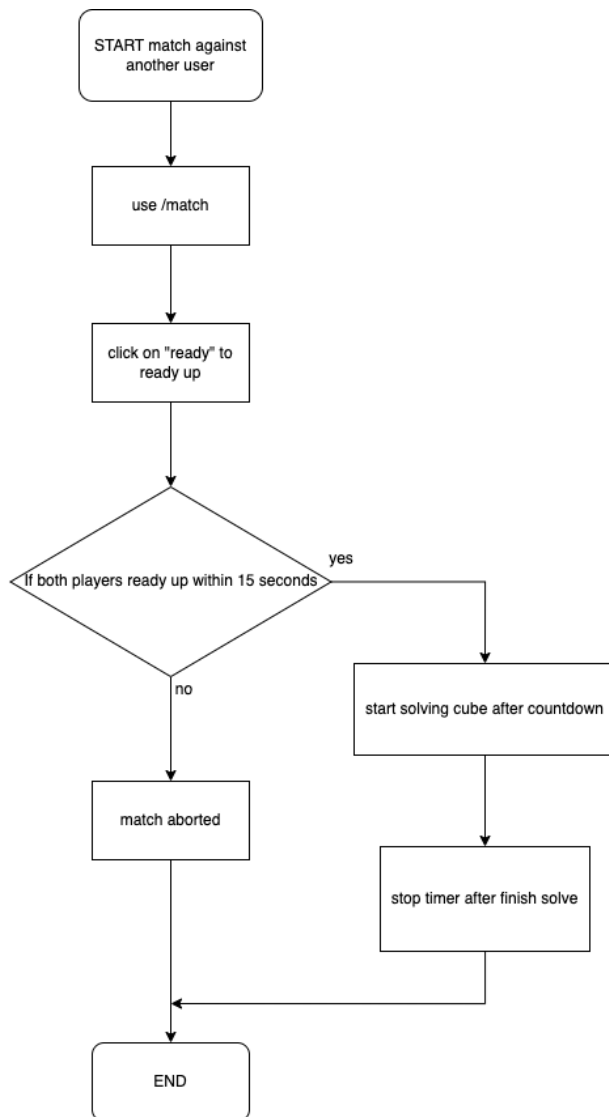

START timing virtual solve

use /virtual

click "start"

use /move to solve the cube

if cube not solved — yes

no

timer ends

END

Figure 7: flowchart for virtual solves

Figure 8: flowchart for match

Test plan

Success Criteria Tested	Action to be Tested	Test Method
1	Help command provides instruction	Use /help without argument and /help on some command. Check whether the description matches the functionality of command
2, 3	Scrambles generated scrambles the cube thoroughly and is correct	Run commands /2x2, …, /5x5 and check if the cube looks thoroughly scrambled and that the cube matches the image displayed. Run /angle to check whether the angle of perspective could be changed
4, 7	Times user's solves and can add penalty	Run /3x3, press start time. Then press stop time when the solve ends. Check whether the time of solve is similar to the length of video. Try adding OK, +2, and DNF penalties to the solve.
6	Challenge other users	Run /match against my other discord account. Check whether the other account can accept/decline the challenge
8, 9	Display most recent solves and statistics	Do 12 solves then run /list and /stats. Check whether the ao5, ao12, and mo3 are correct
10, 11	User can manually add and delete times	Run /addtime and /delete. Then run /list to check whether the time is added/deleted
5	User can play with a virtual cube	Run /virtual then use cube notation to solve the cube. Check whether the virtual cube turns according to the cube notation and can detect invalid notation