

System Tests

Note: For each scenario, the first step you should take is to invite the Discord bot to your server if you have not already done so.

Sprint 1

User Stories

- As a non-technical user, I want to be able to easily setup and use the discord bot
- Scenario:
 1. Use link of discord bot to get to the discord pop up and choose which server to add bot to
 - a. Link is in README at <https://github.com/ryanchowdev/RoommateHelper>
- As a roommate, I want to be able to schedule tasks to be done daily/weekly/hourly or just once
- Scenario:
 1. Go to the channel where you want the schedule to be
 2. Input command for scheduling a message once every minute (lowest time unit)
 - a. `=schedule m 1 "Take out the trash"`
 3. Confirm that the bot is outputting the message based on the time frame given
 4. Delete the schedule
 - a. Use command to `clearSchedule` for `>=1` schedule
 - b. Use `=getSchedule` to get a list of schedules and ids for each one
 - i. Then `=deleteSchedule <id>` to delete the specific schedule
- As a renter, I want to be able to divide expenses and bills among everyone in the household
- Scenario:
 1. Go to the channel that you will use for the Discord bot
 2. (optional) Use the calculator function to find out how much each person owes, if the total is the same for each person, with the following command:
 - `=calc <amount> / <people>`
 3. Record the debts with the following command. Repeat for each person that has debt:
 - `=debt <name> <amount> <note>`
 - The note parameter is optional.
 4. Check the debts with the following command:
 - `=checkdebt`
 - User should see a list of all current debts.
 5. If you want to change the debt amount, use:
 - `=changedebt <name> <amount>`
 - OR to change the note, use:
 - `=changenote <name> <note>`
 6. When the debt has been paid, remove it with the following command:
 - `=removedebt <name>`
 - OR to clear all debts, use:
 - `=cleardebt`

7. Verify that the debt has been removed by checking again with the following command:
 - `=checkdebt`
 - User should see that the selected debts have been removed. If there are no more debts, the user should see the message "No current debts."
- As an user, I want to get the weather forecast to plan how I dress
Scenario:
 1. Go to the channel you will use for the Discord bot
 2. If this is your first time using the weather command, set your city with the following command:
 - `=setCity <city>`
 3. If you want to change the units system from the default setting of imperial to metric or back from metric to imperial, do so with the following command:
 - `=setUnits imperial` OR `=setUnits metric`

Sprint 2

User Stories

- As a Discord User, I do not want to re input all the information on debts/scheduling everytime the bot goes offline
Scenario:
 1. Since we are using a Database, our data persists upon reboot. To test this, first go to the channel that you will use for the Discord bot.
 2. Now, record some information that should persist, such as a debt or schedule. For example:
 - `=debt john 150 bills`
 3. Check that the information was recorded.
 - `=checkdebt`
 - User should see that john owes \$150 for bills.
 4. Kill the bot.
 5. Restart the bot.
 6. Check that the information persists.
 - `=checkdebt`
 - User should see that john still owes \$150 for bills.
- As a roommate, I want to establish rules and access them easily
Scenario
 1. Go to any text channel
 2. Record the rule you would like to insert
 3. Insert `addRule` command
 - a. `=addRule Don't smoke in the house`
 4. Check if it has been added with the `getRules` command
 - a. `=getRules`
 5. If you want to clear the ruleBoard use the `clearRules` command
 - a. `=clearRules`

- As a friend, I want to be have polls in order to decide with my friends on various decisions (ie: where to eat, what movie to watch, etc)

Scenario:

1. Go to any text channel
2. Make a poll of the question you want to ask and the options you want to give.
 - a. =poll 3 "When should we watch the movie?" 7:00 8:00 "9:00 PM"
 - b. User should see a poll with the question "When should we watch the movie?" and 3 options to react to: 7:00, 8:00, and 9:00 PM.
3. As another option to deciding on various decisions, use a coin flip to randomly decide between two options.
 - a. =flip
 - b. Example: Deciding whether to go to Restaurant A or Restaurant B, where heads or tails will decide on which option to go with
 - c. User should see a text response of either "Heads" or "Tails" with an image of the corresponding result.

Sprint 3

User Stories

- As a discord user, I want to have an emergency music bot if the others get taken down again

Scenario

1. Join a discord voice channel
2. Have the bot join through musicJoin
 - a. =musicJoin
3. Have the bot play music through the playMusic command
 - a. =playMusic <https://www.youtube.com/watch?v=XBSZ5yvB5mo>
4. Pause the music with musicPause
 - a. =musicPause
5. Resume the music through musicResume
 - a. =musicResume
6. Queue up another song with playMusic
 - a. =playMusic https://www.youtube.com/watch?v=vtSXNQSA_t0
7. Skip the current song with musicSkip
 - a. =musicSkip
8. Make sure the next song plays and then use the musicSkip command again

- As an organizer, I want to have scheduling messages in a specific channel

Scenario

1. In any channel use the restrictChannelSchedule command
 - a. =restrictChannelSchedule channel
2. See if scheduled messages appear only in the channel
3. Remove the restriction with
 - a. =removeChannelRestrict

Sprint 4

User Stories

- As a student, I want to look up locations of interest near me on Google Maps quickly and easily (for example, pizza or barber shop).

Scenario:

1. Go to any text channel
 2. Query the bot with your location of interest
 - a. =places <location of interest>
 3. Click the link that the bot replied to you with
- As a technical user concerned about privacy, I want to host my own instance of the Discord bot.
 1. Go to the project's Github page (<https://github.com/ryanchowdev/RoommateHelper>)
 2. Clone the project and install the necessary dependencies
 3. Create a .env file with your API keys
 - a. Openweathermap.org API key
 - b. Discord Bot API key
 4. Run the program
 - a. python3 main.py