8ball.html

styles.css

```
ChatGPT Experiment {
  [Team Canary (21)]

   if (You = “Software Engineer”):
       You want to know about what we have to say

}
```

8ball.html

styles.css

Table Of ‘Contents’ {

}

8ball.html

styles.css

Our ‘Team’ {

Van
Team Leader

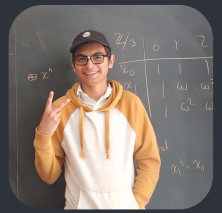
Uncle Sam
Team Leader

Matthew
Planner & FE

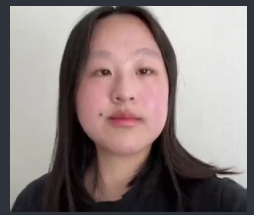
Gil
Back-End Dev

Justin
Designer & FE

Hayden
Front-End



Francisco
Back-End Dev



Michelle
Front-End

Jun
Back-End Dev

Minh
Back-End Dev

}

8ball.html

workshop.css

Our Interaction{

Group1: Application development (Gil, Francisco, Jun, Minh, Sam)

We requested the creation of a Magic Eight Ball app and gave simple and detailed prompts for ChatGPT to follow. ChatGPT is sometimes successful in fixing bugs encountered in the app. An example of the failed trial was that it was initially unable to create a functioning app until we asked it to fix the JavaScript code. It is not predictable in terms of development.

}

8ball.html

workshop.css

Our Interaction{

Group2: Styling and finishing touches (Justin, Matthew, Michelle, Hayden, Van)

We only made significant changes to the code in two instances: when we decided to edit the animation in the CSS themselves and when they couldn't figure out how to add a button feature. We asked ChatGPT a question to understand how the HTML and JavaScript interacted, and were able to combine the generated code with the answer to create a cleaner and concise code and function.

}

8ball.html

workshop.css

```
What { WE
thought;

}
```

8ball.html

workshop.css

Our Thoughts {

Group1:

Concerns about code quality and usability of ChatGPT for CSS modifications: Users have experienced some difficulties when using ChatGPT to modify CSS code. This includes concerns about code quality and overall usability.

Overwriting existing code during revisions: ChatGPT sometimes overwrites existing CSS code when making changes. This can lead to loss of previous styling, which can be frustrating for users.

}

8ball.html

workshop.css

Our Thoughts {

Group1:

Manual copying and pasting as a workaround: To avoid losing important code, users have resorted to manually copying and pasting the code before making revisions. This extra step can be time-consuming and inefficient.

Difficulty referencing specific code: There have been instances where ChatGPT struggled to understand which piece of code was being referenced. This confusion necessitated creating a new chat and repasting the code, which can be inconvenient.

}

8ball.html

workshop.css

Our Thoughts {

Group1:

ChatGPT's value in providing coding assistance: Despite these issues, ChatGPT has proven useful in offering coding support, demonstrating its potential as a helpful tool for developers.

Suggestions for improvements: To address these challenges, several improvements could be made, such as providing clearer documentation for users, and enhancing the handling of existing code during revisions. This would ensure better code quality and a more user-friendly experience.

}

8ball.html

workshop.css

Our Thoughts {

Group2:

Improved JavaScript code quality in Magic Eight Ball app: The JavaScript code in the Magic Eight Ball application shows a positive trend in quality, with each iteration leading to better code. The final attempt demonstrates a high level of quality and includes helpful comments.

Inconsistent HTML code: On the other hand, the HTML code exhibits inconsistency across iterations. This is evident in the use of CSS classes instead of ids and varying metadata provided throughout the attempts.

}

8ball.html

workshop.css

Our Thoughts {

Group2:

Script tag placement varies: Another inconsistency observed is the placement of the script tag. In some attempts, it is located at the bottom of the body tag, while in others it appears in the head tag.

Stylistic choice in the ninth attempt: In the ninth attempt, a paragraph tag is used for the fortune output instead of a div. This stylistic choice was noted as it differs from previous attempts.

Absence of tests and well-engineered aspects: There is no mention of tests or other components typically associated with well-engineered code. The absence of these elements may indicate areas for further improvement.

}

Debian.html

workshop.css

```
That's IT; {

    if you = ‘have any questions’:
        Then ASK
    else:
        print(“Thank you for watching!”)

}
```