**Powerups**

**Possible solution 1 - point saver:**

One feasible powerup would be a point saver. Each player would be allowed to use it once in their entire game. It would work by allowing the player in the answering stage of a question to activate it, upon activated whatever points that player will obtain for that question would be halved. So, if they took a while to answer or answered wrong and got 10 points, it would only be 5. if the result of the halving is not an integer then we just round up/down.

**Implementation:**

Overall, very simple on the logic side. Would require an alteration to the database maybe, each player would have a record of which powerups they have used and have not. Could also be server-side, as it's just a binary check. Overall, the necessary coding would be applying the point saver, halving the points, and then marking that user’s powerup as used

**Problems:** Might be balancing issues involved, could change it from halving to just negating 2 point etc. Will have a clearer idea once the scoring system is finalised.

**Feasibility:** Very simple code overall, hardest bit to implement would be checking a players available powerups, but this would have to be done anyway. 8/10

Note:10/10 is most feasible

**Possible Solution 2 - Answer hider:**

Another very easy and feasible powerup would be an answer hider. Like the point saver it would work in the same way as in only one per user per game. Using the powerup would hide one of the incorrect answers from the user, or even make the box around the answer appear red to indicate it is not correct. It would stay like this for the rest of that round and then return to normal at the start of the next.

**Implementation:**

The code could be done in the front end, as its more to do with visibility than internal logic. We would just set the visibility of a wrong answer to false or change the color around that answer to indicate it is false. Once again, we could track the players powerup availability in the database or an object in the backend so we can know whether they have already ‘spent’ it.

**Problems:**

**Feasibility:** Once again very simple code, maybe even more simple than the point saver. 9/10

**Possible Solution 3 - Probability:**

A probability powerup would allow the player to view the probability of each answer for a question. When the player uses the powerup every answer will display a percentage value attached to it. This value will indicate what other players have previously chosen as their answer. Overall, it gives the player a rough hint as to the correct answer, but the answer with the highest percentage can still easily be wrong

**Implementation:**

A bit trickier than the previous ones. Would require us to store a percentage attached to every answer/question in the database and dynamically update that percentage every time an answer is chosen by any player. We would then need to pass in and display that percentage on the front end which would potentially require changes to the API schema.

**Problems:**

**Feasibility:** A lot of work for a powerup. It is doable but in terms of hours put in vs output the two previous powerups seem a lot more lucrative. 3/10

**Possible Solution 4 – Double Down:**

Double down would be a powerup that would allow the player to maximize their points or spoils on a correct answer. Basically, once the powerup has been activated and the player chooses the correct answer, they obtain double the rewards but if they answer wrong, they receive double the penalties. Provides a cool risk/reward dynamic to the quiz.

**Implementation:**

Very similar implementation to the point saver. Would need an if case for whether they got the correct answer or a wrong one. On the correct branch it would half their points for that question and on the wrong branch it would double their points.

**Problems:**

Like the point saver there is possible balancing issues attached to this question but more so to This as a skilled player might shoot too far ahead of the rest of the lobby. Will have a better idea after some testing.

**Feasibility:** slightly more complex than the point saver code but overall, very simple, 7/10

**Possible Solution 5 – Second Chance**

Second chance would be a powerup that allows the player to take two chances to pick the correct answer. The powerup would only be able to be used before the first answer has been chosen. If the player selects a wrong answer while Second chance is activated no penalty is given, instead they can try again to pick the right answer. If they pick the wrong answer again then they receive penalty as normal and if they select the correct, they are awarded as normal.

**Implementation:**

While it seems relatively simple there could be some complications when it comes to submitting the players answer. Once the powerup is activated there would have to be some sort of clone of the normal if statement that checks for the correct answer with slightly altered functionality. If the players answer is right then the code would run as normal, if it is wrong then ideally, we would restart the if statement. On the second run through if the players answer is wrong then it will be the normal penalty. The second run through could even just return to the original if statement for checking the correct Answer.

Another way of implementing would be to alter the current answer checking functionality. When the player inputs a wrong answer, we would check if a boolean like ‘SecondChanceOn’ = true. If true, return to waiting for an answer from the player and set ‘‘SecondChanceOn’ to false.

**Problems:** As written in the implementation there could be some complications when it comes to waiting for the player to select an answer again.

**Feasibility:** A bit more complicated than some of the powerups but still very achievable, 6/10

**Possible Solution 6 - Hint**

The hint powerup upon use would reveal to a player about the current question. the hint would supply a small bit of information that would make identifying the correct answer easier.

**Implementation:**

The implementation would be very straightforward but time consuming. Every question in the database would have a hint attached and that hint would be sent to the front end with the question. Upon activating the hint power up that hint's visibility would just be set to true

**Problems:**

Having to go through every question and then come up with a consistently helpful and suitable hint would be very time consuming. We would also have to change some of the API schema for passing the hint alongside the question

**Feasibility:** very straightforward addition but also heavily time consuming, 3/10

Independent of what powerups we choose we will have to implement some sort of system that tracks each players availability of each powerup (if they can use it or not). To keep it simple we could make it they have one use per game, or we could implement some sort of way to earn use of powerups in game which could be through streaks or awarding them to losing players. Another idea would be each round a player has one random powerup they can use for that round.

Ideally, I think we should choose 2-4 powerups to use in the game, any more could get very cluttery and hard to balance.