

Goal:

Develop an Asp.Net Core application, that will handle the next situation:

3 players (clients) are in the game. One player creates a room (lobby) and is treated as a host. Other in-game players attempt to join the room, but only 2 players (including the host player) can be in the room at the same time. Whenever the room is full (2 players joined the room, total 2 players), the game will start automatically.

Each player has a "Health=default(10)" parameter. During the game players' health randomly decreases by 0-2 each second until one of the players dies (health<=0). Room must have only one winner: case when one of the players is alive and another player dies, afterwards the room closes, reporting status to the players and saving it into the database. No draw is possible.

Prerequisites:

1. Use asp.net Core as backend technology.
2. Use EF code first for database handling.
3. Consider client logic as an isolated blackbox;
4. Provide minimum endpoints that will serve the task.

What is important:

1. Code architecture.
2. Project structure.
3. Strict following to the task requirements, as it was a real project.
4. Restful knowledge
5. Endpoints, cpu & memory performance.
6. Your ability to quickly learn and catch new things on the fly.

What is NOT important:

1. Project type/model.
2. Packages used
3. Database used, db format.
4. Debug/data visualization.
5. Any additional work not related to requirements.

What may be done in addition:

1. Unit tests
2. CQS pattern