Raw Ideas Java Coding Interview

Raw Ideas is building a scoring system for an upcoming Singles Badminton Tournament.

Information on badminton scoring can be found here: <https://en.m.wikipedia.org/wiki/Scoring_system_development_of_badminton>

Pick a Java framework (preferably Play 2 or Spring Boot) to build the following application:

- The input has to be processed through an implementation of the 3 × 21 rally point scoring system to evaluate the result

- The evaluated result should be persisted using Hibernate

- No UI required.

- Sample score input:

 {

    "scores" : [

        {

            "id" : 1,

            "scorer" : "John"

        },

        {

            "id" : 2,

            "scorer" : "Adam"

        },

        {

            "id" : 3,

            "scorer" : "John"

        },

        {

            "id" : 4,

            "scorer" : "John"

        },

        {

            "id" : 5,

            "scorer" : "John"

        },

        {

            "id" : 6,

            "scorer" : "John"

        },

        {

            "id" : 7,

            "scorer" : "John"

        },

        {

            "id" : 8,

            "scorer" : "John"

        },

        {

            "id" : 9,

            "scorer" : "John"

        },

        {

            "id" : 10,

            "scorer" : "John"

        },

        {

            "id" : 11,

            "scorer" : "John"

        },

        {

            "id" : 12,

            "scorer" : "John"

        },

        {

            "id" : 13,

            "scorer" : "John"

        },

        {

            "id" : 14,

            "scorer" : "John"

        },

        {

            "id" : 15,

            "scorer" : "John"

        },

        {

            "id" : 16,

            "scorer" : "John"

        },

        {

            "id" : 17,

            "scorer" : "John"

        },

        {

            "id" : 18,

            "scorer" : "John"

        },

        {

            "id" : 19,

            "scorer" : "John"

        },

        {

            "id" : 20,

            "scorer" : "John"

        },

        {

            "id" : 21,

            "scorer" : "John"

        },

        {

            "id" : 22,

            "scorer" : "John"

        },

        {

            "id" : 23,

            "scorer" : "John"

        },

        {

            "id" : 24,

            "scorer" : "Adam"

        },

        {

            "id" : 25,

            "scorer" : "John"

        },

        {

            "id" : 26,

            "scorer" : "John"

        },

        {

            "id" : 27,

            "scorer" : "John"

        },

        {

            "id" : 28,

            "scorer" : "John"

        },

        {

            "id" : 29,

            "scorer" : "John"

        },

        {

            "id" : 30,

            "scorer" : "John"

        },

        {

            "id" : 31,

            "scorer" : "John"

        },

        {

            "id" : 32,

            "scorer" : "John"

        },

        {

            "id" : 33,

            "scorer" : "John"

        },

        {

            "id" : 34,

            "scorer" : "John"

        },

        {

            "id" : 35,

            "scorer" : "John"

        },

        {

            "id" : 36,

            "scorer" : "John"

        },

        {

            "id" : 37,

            "scorer" : "John"

        },

        {

            "id" : 38,

            "scorer" : "John"

        },

        {

            "id" : 39,

            "scorer" : "John"

        },

        {

            "id" : 40,

            "scorer" : "John"

        },

        {

            "id" : 41,

            "scorer" : "John"

        },

        {

            "id" : 42,

            "scorer" : "John"

        },

        {

            "id" : 43,

            "scorer" : "John"

        },

        {

            "id" : 44,

            "scorer" : "John"

        }

    ]

}

- Sample output:

{

    "id" : 1,

    "players" : [

        "Adam",

        "John"

    ],

    "match\_winner" : "John",

    "match\_scores" : [

        {

            "game\_number" : 1,

            "game\_winner" : "John",

            "game\_scores" : {

                "Adam" : 1,

                "John" : 21

            }

        },

        {

            "game\_number" : 2,

            "game\_winner" : "John",

            "game\_scores" : {

                "Adam" : 1,

                "John" : 21

            }

        }

    ]

}

Notes:  
- Please include a readme with instructions on how to execute and test your code, as well as any assumptions made.  
- Your solution should work for other scenarios not covered by the sample json input i.e. match consisting of 3 games.  
- Your solution should be maintainable and should be easily extendable.  
- You can assume that the input will have the correct number of points per match.  
- You may not use any library that already has the business logic implemented.  
- You may use a Java framework and any unit testing libraries or build tools.  
- When you are happy with your solution and feel that it is of production quality, please compress your source files into a single .zip file.  
- Please do not include any executables or compiled files with your submission.