- Tạo 1 thư mục “ server.js ” có nội dung “

var express = require("express");

var app = express();

var server = require("http").Server(app);

var io = require("socket.io")(server);

server.listen(3000);

const mongoose = require("mongoose");

mongoose.connect(<' chuỗi kết nối của mongo '>, function (e){

if(e){

console.log("Mongo error");

} else {

console.log("Mongo connected successfully. ");

}

});

io.on("connection", function(socket){

console.log("New connection: " + socket.id );

socket.on("disconnect", function(){

console.log( socket.id + " has been disconected. ");

})

})

”

- Tạo 1 file “ Round.js ” trong thư mục “ models ” có nội dung “

const mongoose = require("mongoose");

const AutoIncrement = require("mongoose-sequence")(mongoose);

const roundSchema = new mongoose.Schema({

small\_money: Number,

small\_players: Number,

big\_money: Number,

big\_players: Number,

counter: Number, // 1 -> 60

result: Number, // -1 : waiting, 0: small, 1: big

dice: Number,

dateCreated: Date,

});

roundSchema.plugin(AutoIncrement, { inc\_field : "roundNumber" });

module.exports = mongoose.model("round", roundSchema);

”

- Thêm vào trong file “ server.js ” nội dung “

const mongoose = require("mongoose");

var Round = require("./model/Round");

...

var currentRoundNumber = null;

function createNewRound() {

var newRound = new Round({

small\_money: 1000,

small\_players: 0,

big\_money: 500,

big\_players: 10,

counter: 1, // 1 -> 60

result: -1, // -1 : waiting, 0: small, 1: big

dateCreated: Date.now()

});

newRound.save( function(e){

if(!e){

console.log("New round created: " + newRound.roundNumber);

currentRoundNumber = newRound.roundNumber;

roundCounter(currentRoundNumber);

} else { currentRoundNumber = null; }

});

}

function roundCounter(roundNo) {

Round.findOne({ roundNumber: roundNo}, function(e, round){

if(!e && round != null ){

if (round.counter < 60){

round.counter++;

console.log("Current " + roundNo + ", count: " + round.counter);

round.save((eSave)=>{

io.sockets.emit("server-send-current-round", JSON.stringify(round);

setTimeout(()=>{

roundCounter(roundNo);

}, 1000)

});

} else {

round.result = Math.floor(Math.random() \* 2);

if(round.result == 0){

round.dice = Math.floor(Math.random() \* 3) + 1; // 1 - 3

} else {

round.dice = Math.floor(Math.random() \* 3) + 4; // 4 - 6

}

round.save((eSave)=>{

console.log("Winner is: " + round.result);

setTimeout(()=>{

createNewRound();

}, 1000);

});

}

} else { }

})

}

createNewRound();

io.on( ... )

”