

Steg 1.

Request kommer till frontcontrollern -> routern initierar sig -> Data om metod o path hämtas från request -> funktion i yatzy anropas.

Steg 2.

Sessionen startas om. Databas tabell yatzy nollställs.

Funktion skapar ett objekt "game" som sparas i sessionen, game skapar ett nytt objekt diceHand som kastar 5 tärningar genom klassen dice som skapar 5 tärningar. Game hämtas från session variabeln i viewn yatzy.php för att värdena ska kunna skrivas ut i viewn.

Steg 3.

Efter tärningarna kastas skrivs de ut grafiskt på sidan genom dess properties i en twigfil. Tärningar som ska sparas inför nästa kast sparas och hämtas genom postformuläret. En jämförelse mellan hur många tärningar som sparas och max antal tärningar (5) görs och differensen blir antal tärningar att kasta nästa gång.

Steg 4.

En metod håller reda på när 3 kast har gjorts. Då ska spelaren spara sina värden, spelaren kan också spara sina värden innan 3 kast. Värdena skickas genom ett postformulär.

Steg 5.

När spelaren sparat sina resultat genom formulär skickas datan till en funktion som avgör vilket värde spelaren vill spara, exempelvis ettor. En metod bestämmer räknar ut hur många ettor det finns i den sparade spelhanden, detta hämtas från en metod som kontrollerar de sparade tärningarna och de tärningar som användaren senast kastat. Resultatet skickas till tabellen yatzy i databasen i sqllite. I en databasfunktion sparas datan genom att updatera resultatet för den rad det gäller. Id och en query skickas med beroende på var det ska sparas. Sedan nollställs antal kast som gjorts. Spelrundan ökar med 1 och nästa spelare hämtas ur en array.

När spelarna har spelat 15 rundor var avslutas spelet genom att för varje runda kontrollera whatRound. När den överstiger ett visst antal rundor (15 rundor * antalspelare) sparas det slutgiltiga resultatet i en databas tabell highscore.