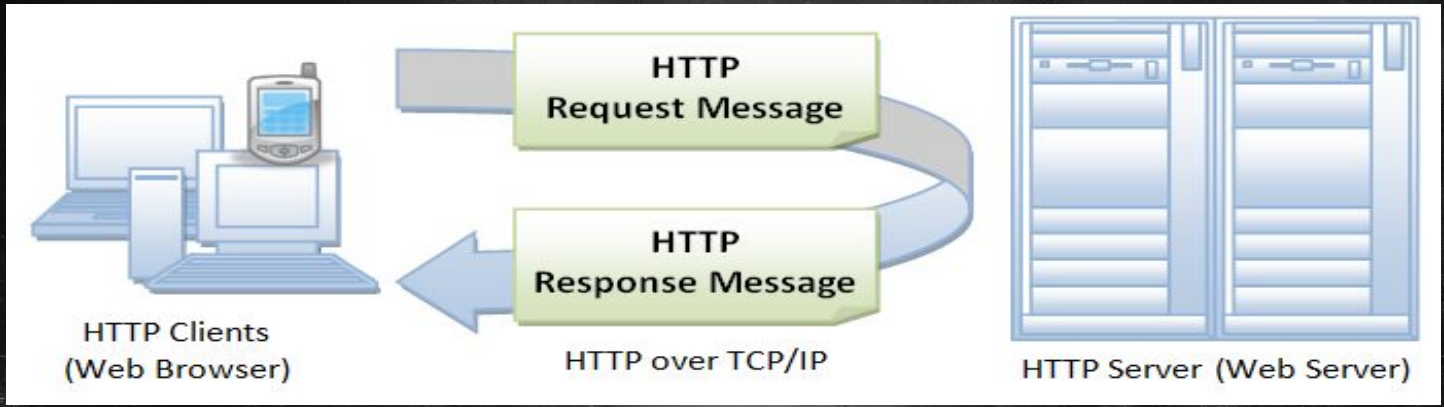
Explaining the HTTP-Request-Response Cycle

Kobe

Taylor

What is HTTP?

- Hypertext Transfer Protocol
  - Defines how messages are defined and formatted between a server and a client
  - Is stateless
    - Each request is independent
- Key Terms
  - Client: The app making the request
  - Server: The system that processes the request and returns a response
  - Request: A message sent by the client, usually asking for data
  - Response: The reply from the server with data or status codes
- Cycle:
  - Client → HTTP Request → Server → HTTP Response → Client



HTTP Request

1. Method - GET, POST, PUT, DELETE → Get Request
2. Endpoint - Route on server your targeting → "api/WorkoutLogger/GetWorkout"
3. Body (if POST or PUT, etc) - The actual data being sent → WorkoutName, etc
4. Format - "PostAsJsonAsync" → Sends it as JSON which our ASP.NET can deserialize into a Workout Object

Constructing the HTTP Request

```
async Task LoadWorkouts()
{
    workouts = await Http.GetFromJsonAsync<List<Workout>>("api/WorkoutLogger/GetWorkout");
}
```

HTTP Response

Endpoint

This endpoint will return all saved workouts each with a list of their own exercises

Data Returned

Returning a workout with a list of exercises, using a DTO to specifically interact with our front end to bind with our razor components