**HTTP Requests and Responses**

1. Client-Server Architecture
2. Verb, uri, http version, request header, request body
3. The body is optional
4. Startline,header and body
5. 400 and 500
6. Get and post
7. POSTS
8. Body
9. Header

**Curl**

1. supports a number of different protocols and gives your access to the response headers.
2. -x --request
3. -H
4. -I
5. OPTIONS

**Sessions and Cookies**

1. Set-cookie
2. GET /cart HTTP/1.1

**Example HTTP Requests and Responses**

**HTTP Request**

1. POST
2. Upgrade-Insecure-Requests: 1
3. Yes connection is ‘keep-alive’
4. username=Barbara&password=password is being sent

**HTTP Response**

1. 200 or OK
2. Apache
3. Yes
4. Page information
5. Strict-Transport-Security: max-age=31536000; includeSubDomains

X-Content-Type: NoSniff

X-Frame-Options: DENY

X-XSS-Protection: 1; mode=block

Monolithic and Microservices

1. services
2. APIs
3. Scalability and resilience

Deploying Testing a container set

1. Docker compose
2. Yml

Databases

1. Select statement
2. Insert
3. It would delete all the data