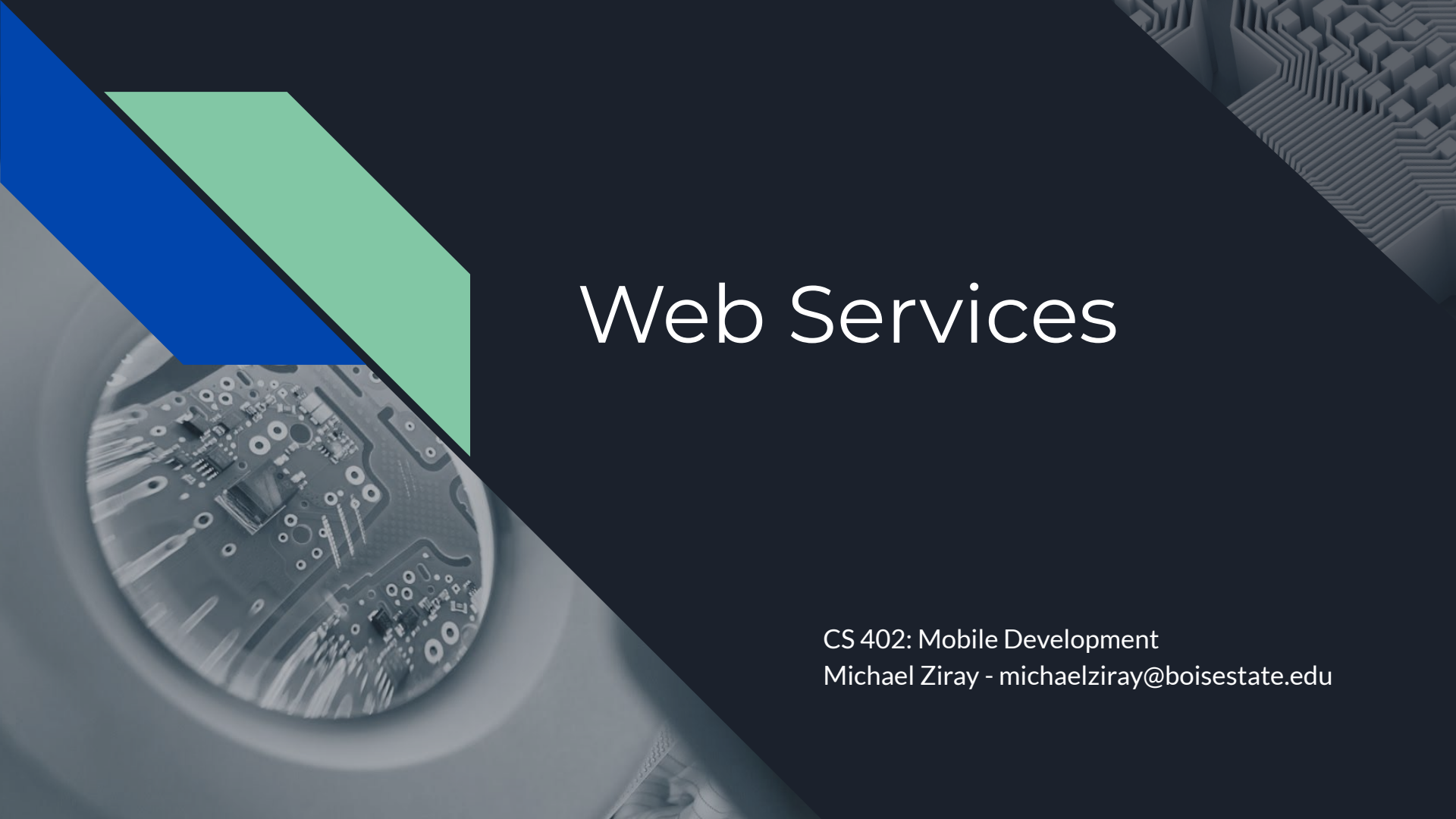

Web Services

CS 402: Mobile Development
Michael Ziray - michaelziray@boisestate.edu

JSON

JavaScript Object Notation

JSON

Stands for JavaScript Object Notation

Is essentially just a JavaScript object

Evaluates to JavaScript

Why JSON?

Web services should be agnostic to the clients they're serving (browser, mobile, watch)

XML is clunky and hard to parse

JSON can be consumed by JavaScript web apps for free

Easily parsed by Android or iOS natively

JSON Example

```
{
  "glossary": {
    "title": "example glossary",
    "GlossDiv": {
    "title": "S",
    "GlossList": {
        "GlossEntry": {
            "ID": "SGML",
            "SortAs": "SGML",
            "GlossTerm": "Standard Generalized Markup Language",
         }
      }
   }
}
```

ReST

Transferring data across a network

Terminology

Request - Any time you send information to the server

Response - The return information from the server, in response to the original request.

ReST

REpresentational State Transfer

Request methods are:

PUT, POST, GET

DELETE, PATCH, HEAD, OPTIONS

GET

Gets a resource from a URI

Most webpage requests use GET

Web browsers: GET -> http://www.google.com

PUT

Puts or replaces a new resource onto the server at a ***specified*** location

Often for updating a record, but not always.

Updating a user's name:

i.e. /api/user/123/?name=new%20name

POST

Posts information to a server

Sends information to the server at no particular URI

Creating a new user:

/api/user => {user: “Mike”, pass: “pass”}

Query Params vs Body

Query Params:

/api/user/123/?name=new%20name // Params need to be encoded

Request Body

/api/user => {user: “Mike”, pass: “pass”} // JSON, XML, etc

DELETE

Removes a resource on the server

DELETE /user/123

** A lot of APIs will use GET /user/123/delete or something similar

Testing JSON

http://www.jsontest.com/

HTTP Basic Auth

Sets the username and password in the header of each request. Must be over SSL (HTTPS).

HTTP Basic Auth

```
let credentials = "username:password"

let byteData:String = Base64.encodeToString(credentials.encodeToByteArray(),
Base64.URL_SAFE or Base64.NO_WRAP)
```

HTTP Basic Auth

Converts the string username:password into base64

Header looks like:

```
Authorization: Basic aDHR0cHdhdGNoOmY=......
```

Tokenized Authentication

Instead of sending a Username/Password for every transaction, you send an authorization token that you receive when first logging in.

Send that token with each request instead.

Tokenized Authentication

The server can invalidate that token at any time for security reasons or session expiration.

Ex: auth=”zyxw987cba321” // expires at 5:00 AM Sept. 29, 2019

Ex: auth=”999zyxw111cba” // expires at 5:00 AM Oct 12, 2019

JSON Web Token (JWT)

https://jwt.io/

JSON Web Tokens are an open, industry standard RFC 7519 method for representing claims securely between two parties.

JWT.IO allows you to decode, verify and generate JWT.