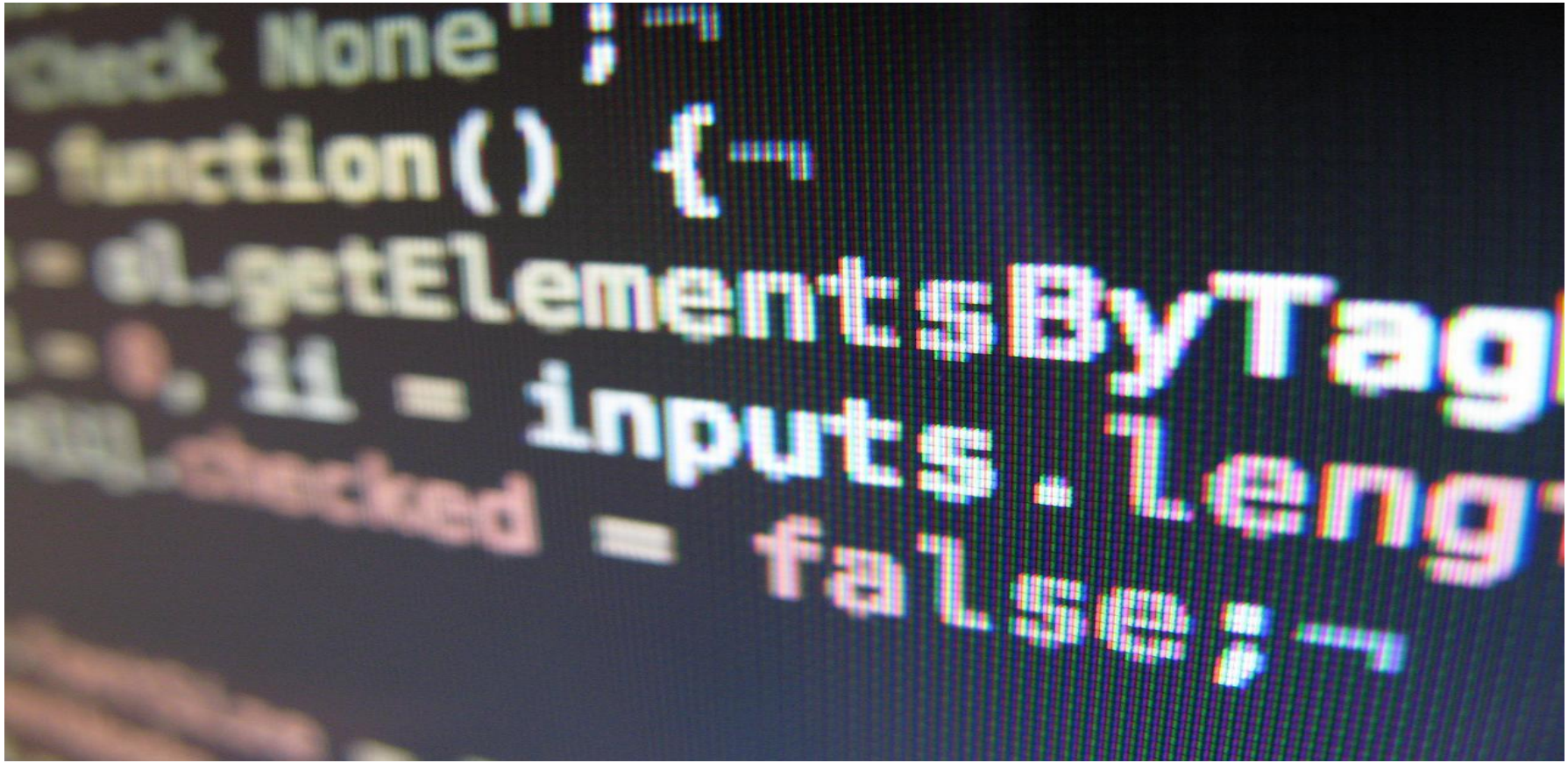
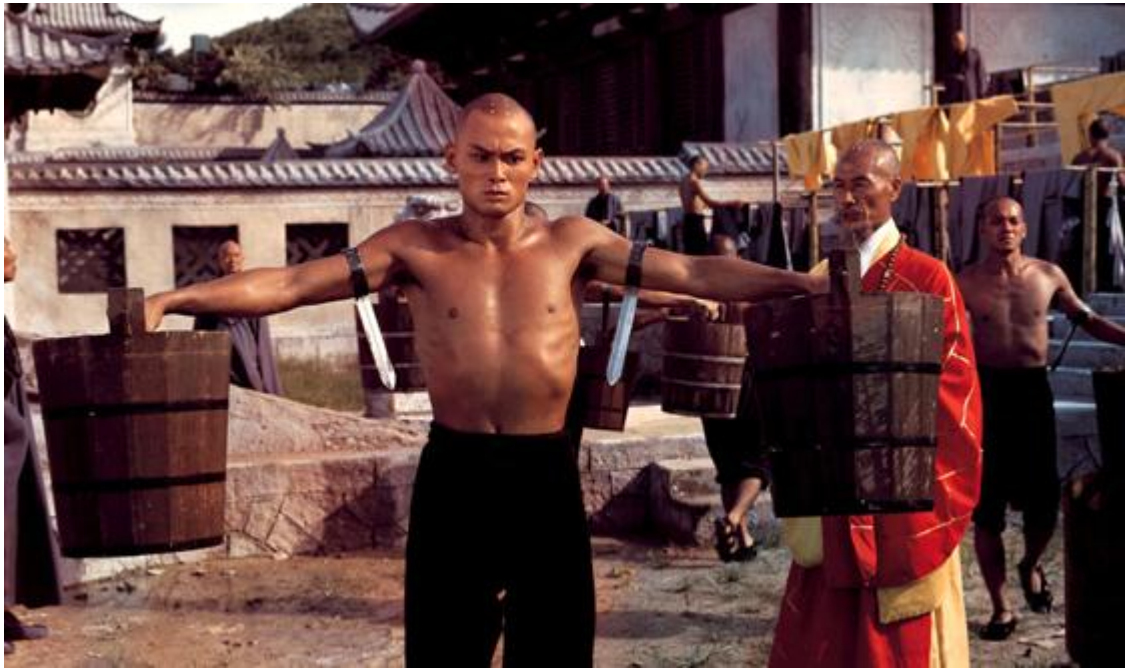
JAVASCRIPT ADVANCED



ICE BREAKERS

Who are you????
Why are you here????
What do you do when you’re not here???
Name your favorite app you used in the last week.

Today your journey to the full stack begins



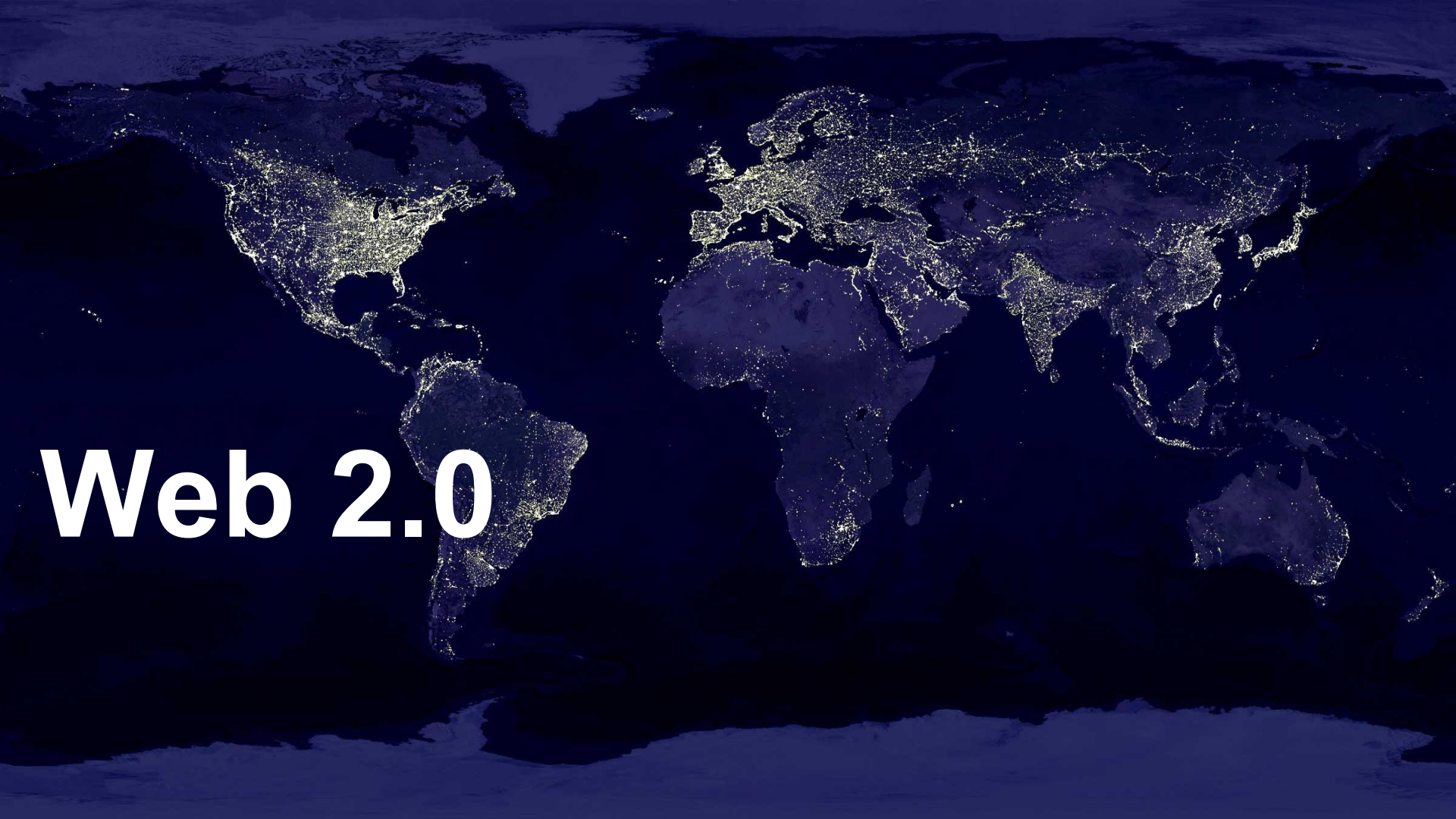
But to know where you are going you must first understand where you are coming from…

To sell your services you must sell yourself

Who are you, why are you qualified to work on this project, and what makes you different

Personal Pitching

- Your Name
- Your Purpose
- Your journey to discover your passion/calling


Web 2.0

What is Web 2.0

Web 2.0 does not refer to any actual change in the infrastructure of the web but rather how it is used.

Examples of Web 2.0

Blogging

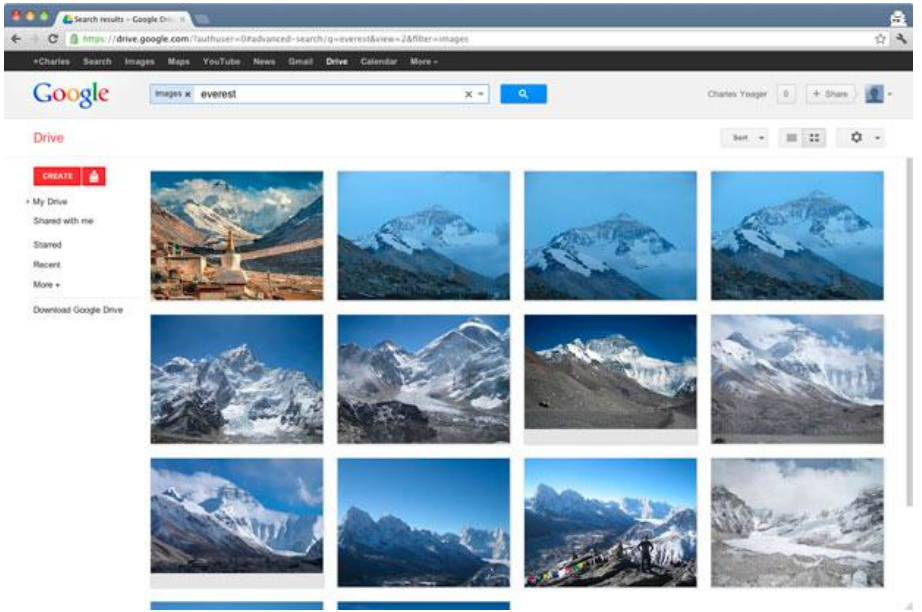
Social Networks

Media Sharing

Wikis

Web Applications

Web 1.0 Vs. Web 2.0



Static - content delivery vehicle
Low Bandwidth

Dynamic Pages
Interactive Applications

Permanent Beta

Sites are always changing based on user behavior
Bug testing, is user generated, and the site is continuously evolving

Features of Web 2.0 Sites

Dynamic Content Generation

Comment and posting functions

Authoring / User Registration

File storage / Sharing

Technologies Behind Web 2.0

Front End

Back End

Databases

The Back End (Server Side)

This is where the websites functionality lives.

Data organization, and transformation
File storage and retrieval
URL mapping and template generation

Model View Control MVC

The server scripting creates and MVC system:

Model: Create a structure for holding the data that is stored in the DB, notified view when changed

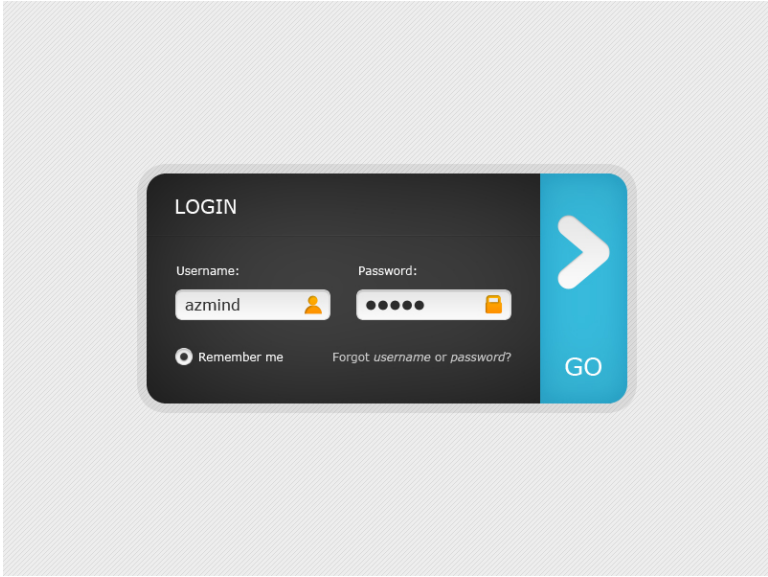
View: Generates dynamic content based on stored data - where user interacts with site

Controller : Applies changes to the model based on user interaction with views and generates views from models based on user requests

MVC by Example

The Following Occurs when you login to your favorite sites.

1. View form fields allow users into input data.
2. The data is received and the controller verifies it is correct.
3. The controller then accesses the model and searches for and pulls in your user info.
4. Then the controller updates the view with your welcome message.



Server Side Frameworks

Python

Ruby

PhP

JavaScript

DataBases

This is where all data is stored and eventually queried and retrieved by a back end framework.

Relational Vs. Non Relational

Relational

Data stored in tables

Better for clearly defined non complex data

Queried through SQL

Non relational:

Data stored as objects

Queried directly

Better for data where relationships are more complex

What to think about when building a WebApp

What is the problem you are solving / for who?

What kind of data are you storing and serving, how would it make sense to be organized

What does the user experience look like, and what tools are best suited to achieve that experience.

What is your goal for the application?

- Monetize with ads
- Generate a large user base
- compile information
- interact visually