Everything about Web Development
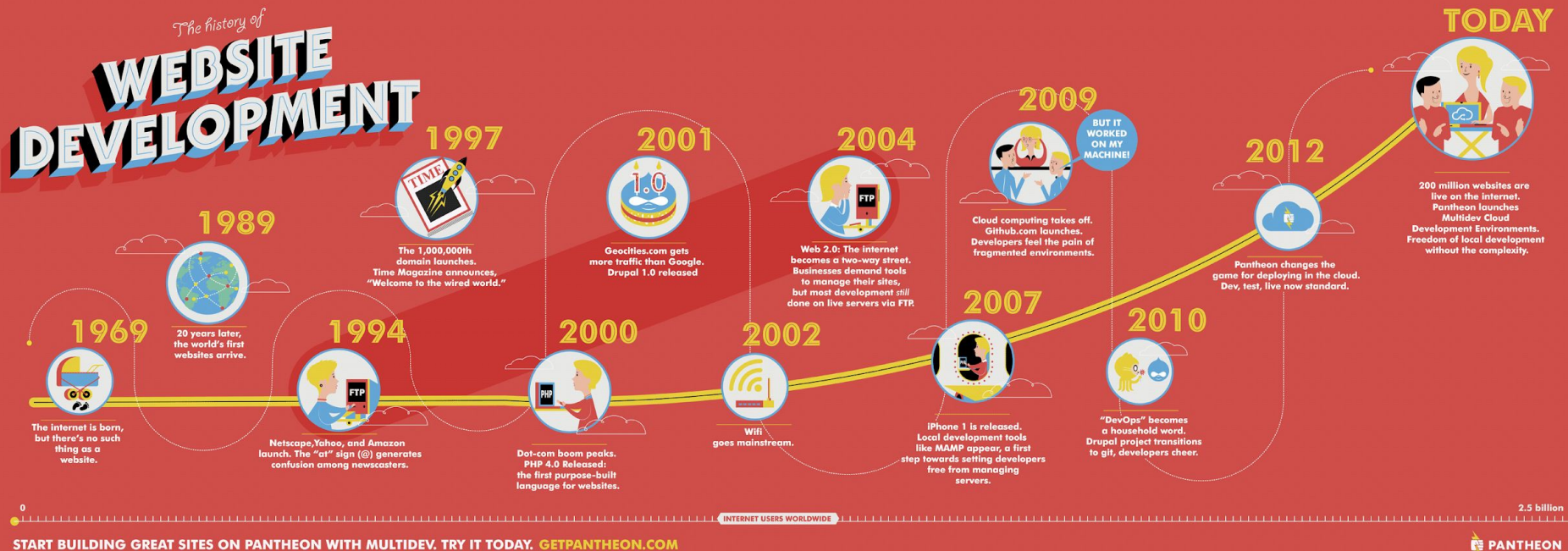
Tanmoy Sarkar
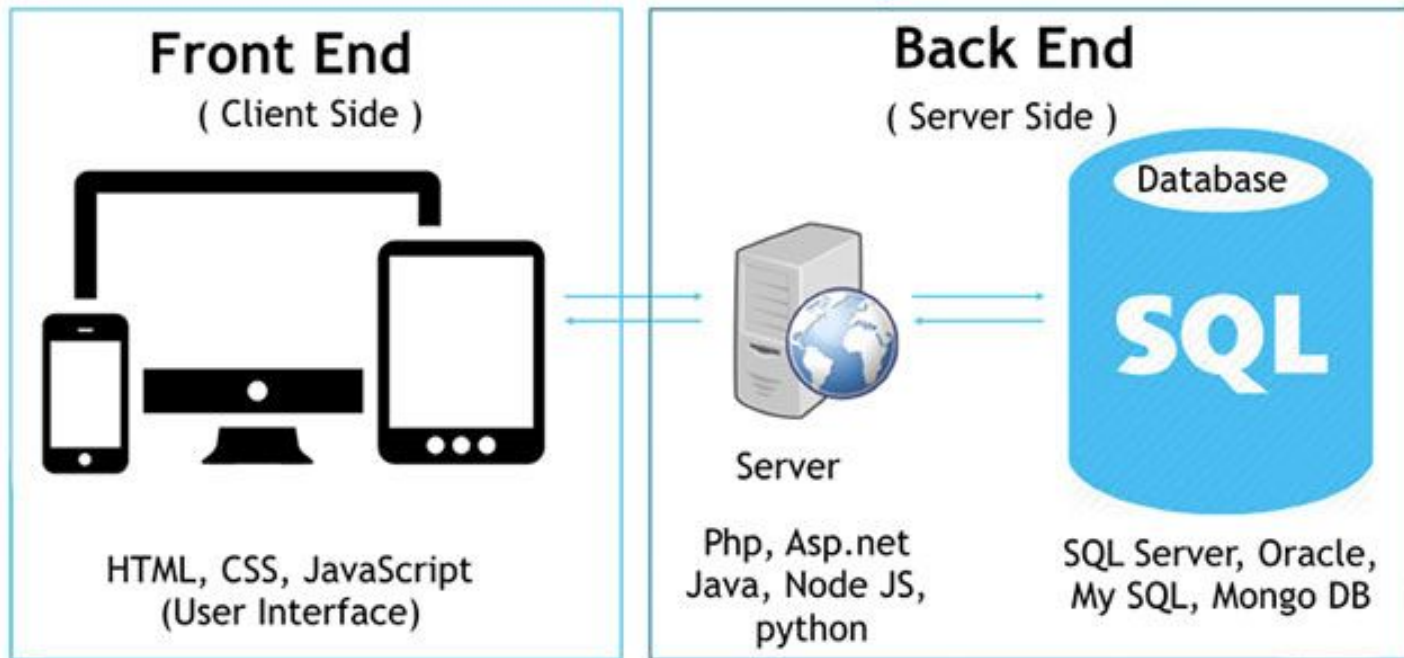
Timeline of web development
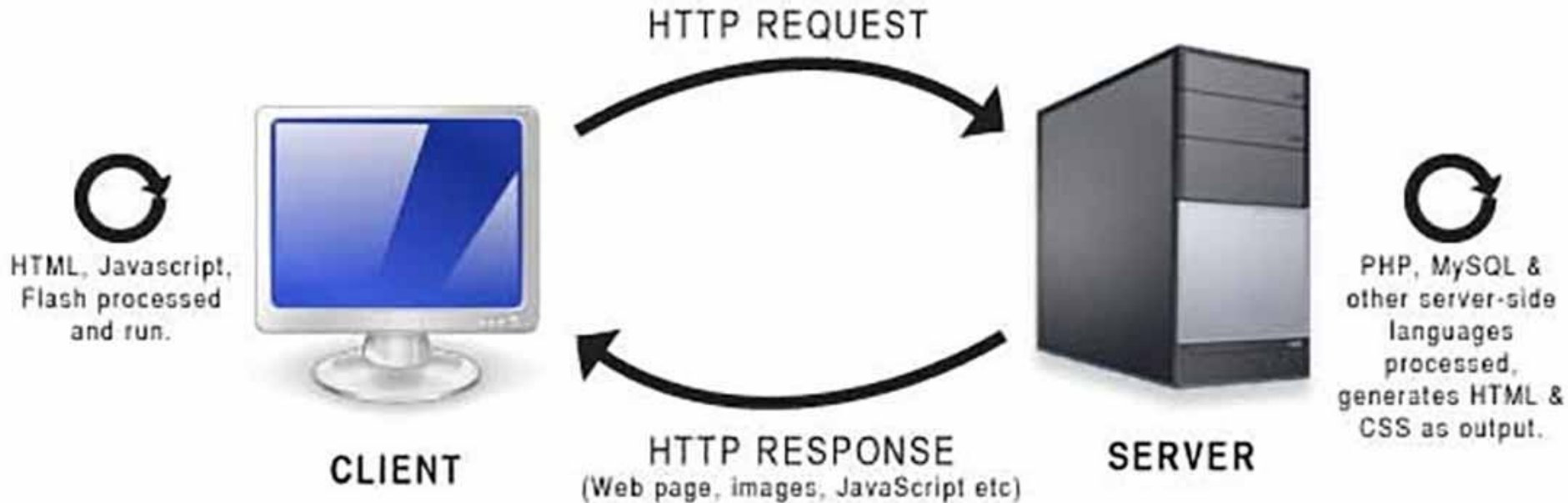

The history of
WEBSITE DEVELOPMENT

TODAY
200 million websites are live on the internet. Pantheon launches Multidev Cloud Development Environments. Freedom of local development without the complexity.

2009
BUT IT WORKED ON MY MACHINE!
Cloud computing takes off. Github.com launches. Developers feel the pain of fragmented environments.

1997
The 1,000,000th domain launches. Time Magazine announces, "Welcome to the wired world."

2001
1.0
Geocities.com gets more traffic than Google. Drupal 1.0 released

2004
FTP
Web 2.0: The internet becomes a two-way street. Businesses demand tools to manage their sites, but most development *still* done on live servers via FTP.

2012
Pantheon changes the game for deploying in the cloud. Dev, test, live now standard.

1989
20 years later, the world's first websites arrive.

1969
The internet is born, but there's no such thing as a website.

1994
Netscape, Yahoo, and Amazon launch. The "at" sign (@) generates confusion among newscasters.

2000
PHP
Dot-com boom peaks. PHP 4.0 Released: the first purpose-built language for websites.

2002
Wifi goes mainstream.

2007
iPhone 1 is released. Local development tools like MAMP appear, a first step towards setting developers free from managing servers.

2010
"DevOps" becomes a household word. Drupal project transitions to git, developers cheer.

0 ——————————————— INTERNET USERS WORLDWIDE ——————————————— 2.5 billion

START BUILDING GREAT SITES ON PANTHEON WITH MULTIDEV. TRY IT TODAY. GETPANTHEON.COM

⚡ PANTHEON

Front-End / Back-End

Web App

mobile App

Server

Data base

File Server

Full Stack Web Development

Front End
(Client Side)

HTML, CSS, JavaScript
(User Interface)

Back End
(Server Side)

Database

SQL

Server

Php, Asp.net
Java, Node JS,
python

SQL Server, Oracle,
My SQL, Mongo DB

Response and Request

_ _ _

HTTP REQUEST

HTTP RESPONSE
(Web page, images, JavaScript etc)

HTML, Javascript, Flash processed and run.

CLIENT

SERVER

PHP, MySQL & other server-side languages processed, generates HTML & CSS as output.

FULL STACK DEVELOPER

Backend

PHP	Perl
Python	Scala
Ruby	Node.js
Java	Docker
C++	MongoDB
.Net	

Frontend

AJAX
Angular JS
Phone Gap
Bootstrap
jQuery
JSON
ReactJS
PWA
SPA

Protocols & APIS

REST
SOAP
API
WebSockets

Database

ProstgreSQL
DB2
MySQL
NoSQI

Platforms

AWS	Linux
Azure	Windows
Google	Andriod
IOS	IBM

Front-End vs. Back-End Development:
Which Side of the Screen Are You?

Is Django full stack?

———

Django is the most popular Python framework for web development. It's a full-stack framework and includes all the necessary features by default instead of offering them as separate libraries.
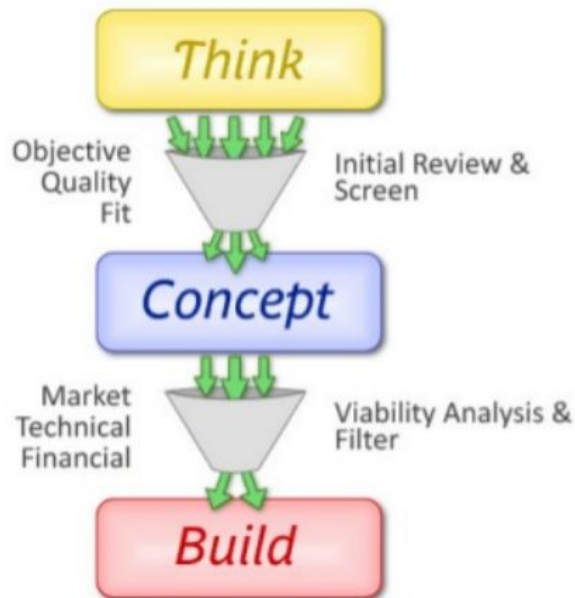
5 BROWSER

4 HTML5 JS CSS3 D3

Leaflet

Javascript

CLIENT

APACHE HTTP SERVER

1 wsgi

django

2 Python

HTTP GET \ POST requests

Users

Data Objects

Requests

3 PostgreSQL

extract

data processing

Microsoft SQL Server

Python

SERVER

```
HTTP Request

    ↓

   URLS

    ↓

Template  ↔  VIEW  ↔  Model

              ↓

      HTTP Response
```
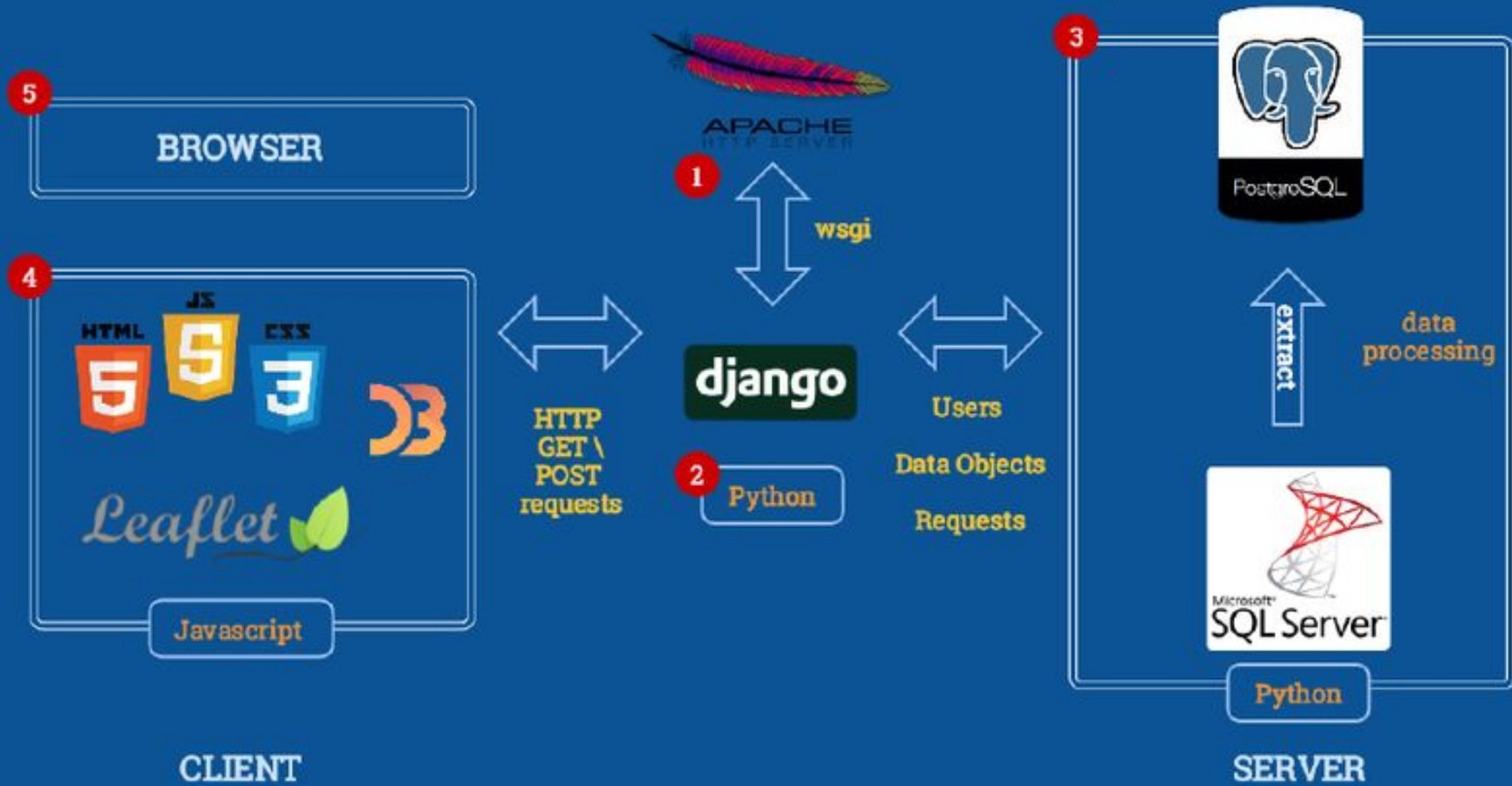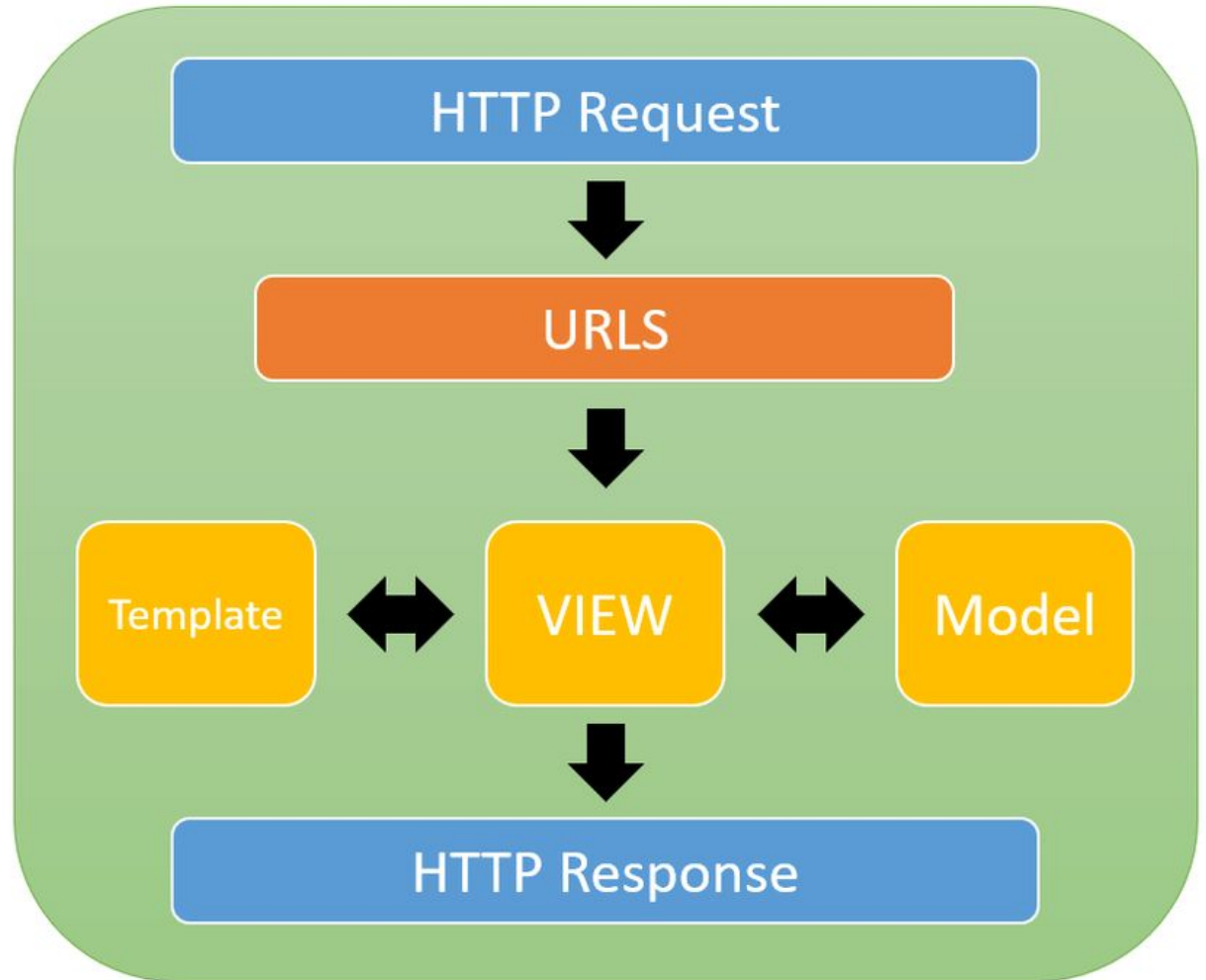
Stages of Web Developement

———

1. Concept development
2. Prototype Development
3. Project Development
4. Testing
5. Deployment
6. Maintenance

Web Development Stages
1 – Concept Development

Write down the following:

- Who is this website for? (the types of users)

- Why am I making it? (what is the problem?)

- What is the most important thing visitors should do on your website.

- Write, read. Write, read. Repeat.

- Sketch (draw) your website outline.

Think

Objective Quality Fit — Initial Review & Screen

Concept

Market Technical Financial — Viability Analysis & Filter

Build

Web Development Stages
2 – Prototype Development

A "broken" version of your website:

- Links work, but don't lead to any content.

- "User Journey" is complete:
  - "As a _____ I want to _____"

- Quick to change and move things around.

Presentation software	Mobile apps	Mobile prototyping applications		
		Hotspot apps	Wireframe / Mockup tools	Prototyping tools
gliffy	AppCooker	APP IN SECONDS	balsamiq	AppSketcher
Keynote	iMockups	FIELDTEST	moqups	axure
OMNI THE OMNI GROUP	Interface 2	inVISION	pidoco	iN Indigo
Visio	mockapp	POP	SOFTANDGUI	proto.io
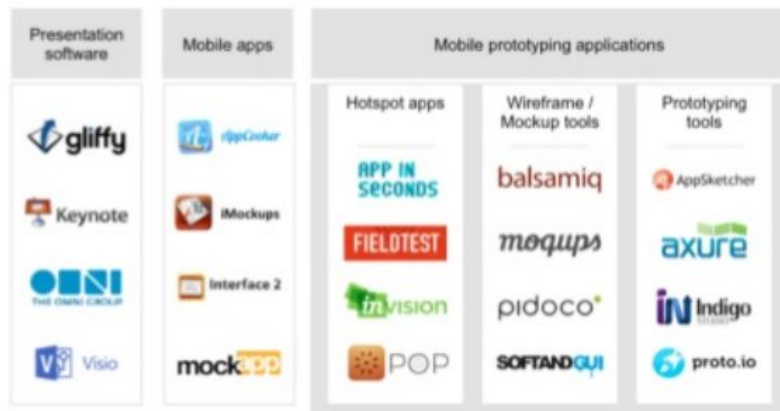
Web Development Stages
3 - Development

Ideally, your requirements are complete

- Work at an *iterative pace*:
  - Develop at the lowest possible work unit

- Use the tool that suits you, and makes you _the most productive_:
  - Text Editor
  - IDE
  - Terminal
  - GUI

Atlassian

Dw

eclipse

JetBRAINS

NetBeans

Bowery

Web Development Stages
4 - Testing

If you don't test; prepare to fail.

Types of testing:

- Functional Test (does the feature work?)
- User acceptance test (is it what the user wants?)
- Unit test (is the code stable?)
- Coverage tests (are all code paths executing?)

Web Development Stages
5 - Deployment

Deployment Process Should Be:

- Documented

- Repeatable

- Predictable

- Rehearsed

- Tools help!
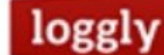
Web Development Stages
6 - Maintenance

Small effort, big rewards

- Enable maintenance by tracking:

  - Successful requests

  - Failed requested

  - Repeated failed requests

  - Key function execution

  - Infrastructure

Tools & Frameworks

USE THESE TO JUMPSTART YOUR DEVELOPMENT

Categories of Tools & Frameworks:
Server Side Development

Categories of Tools & Frameworks
Frontend Development

Categories of Tools & Frameworks Platforms

Dev:Network — Developer Technology Landscape (Version 1.0)

Designed by DevNetwork.com · May 2014

The Internet

A LARGE NETWORK OF COMPUTERS, CONNECTED THROUGH A COMMON NETWORK SPREAD ACROSS A LARGE AREA.

World Wide Web

AN **INFORMATION NETWORK** BUILT ON TOP OF THE INTERNET USING HYPERLINKED DOCUMENTS.

HTML & HTML5

HYPERTEXT MARKUP LANGUAGE. A LANGUAGE CONSISTING OF A SERIES OF TAGS USED TO ANNOTATE DOCUMENTS THAT ARE TO BE LINKED TOGETHER ON THE WEB. THE LATEST VERSION IS 5

CLIENT SIDE

A TERM USED TO DESCRIBE TECHNOLOGY THAT RUNS ON THE DEVICE CONNECTED TO THE INTERNET AND REQUESTING DOCUMENTS FROM A SERVER. EXAMPLE: JAVASCRIPT, FLASH

SERVER SIDE

A TERM USED TO DESCRIBE TECHNOLOGY THAT RUNS ON THE SERVERS CONNECTED TO THE INTERNET PROVIDING ONE OR MORE SERVICES.
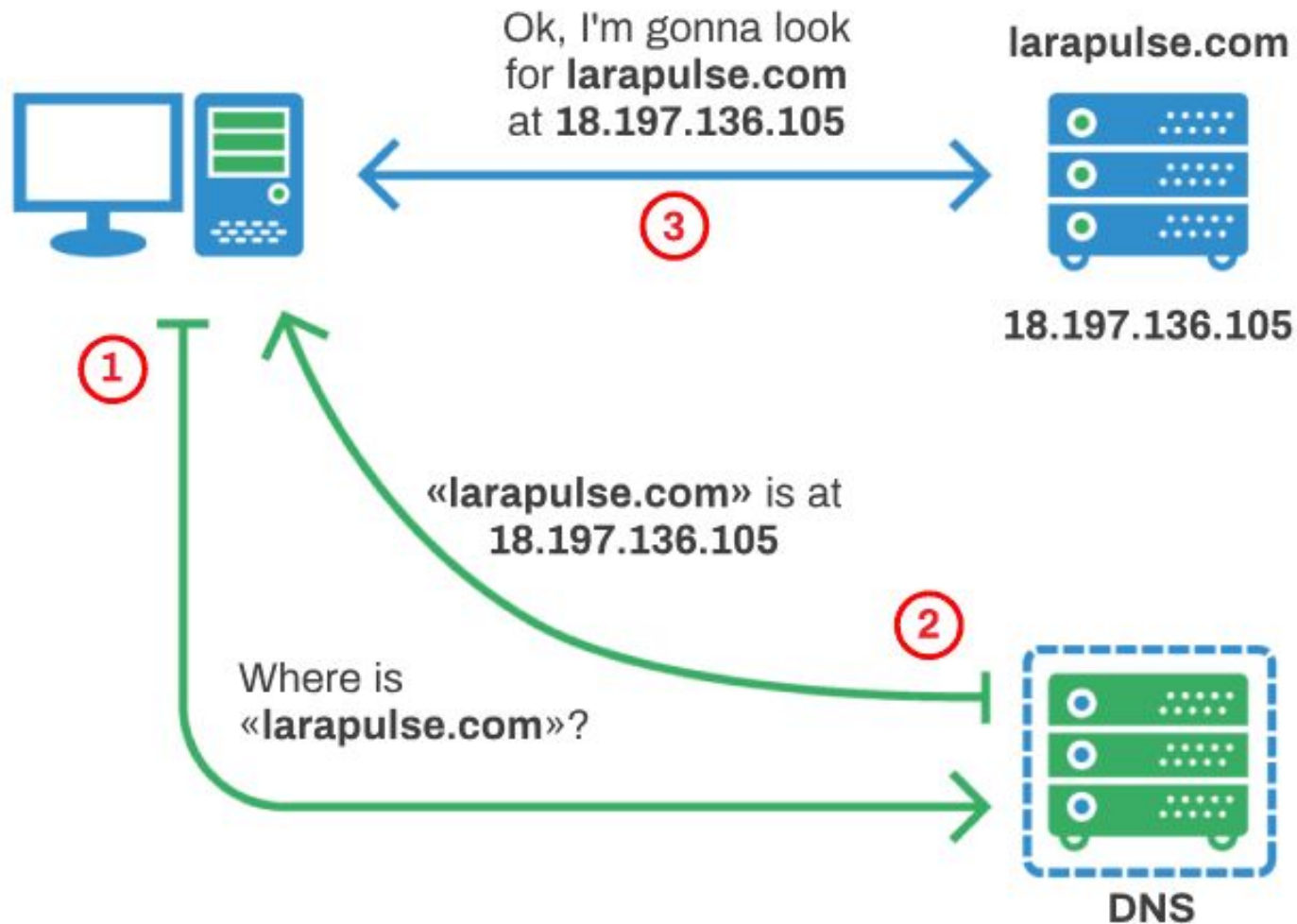
TCP/IP
IPv4
IPv6

PROTOCOLS USED TO COMMUNICATE WITH SERVERS ON THE INTERNET. EACH DEVICE THAT CONNECTS TO INTERNET NEEDS AN IP ADDRESS.

DOMAIN NAME SYSTEM

A DISTRIBUTED SYSTEM DESIGNED TO ASSIGN NAMES TO IP ADDRESSES, IN ORDER TO IDENTIFY RESOURCES ON A NETWORK.

DNS

– – –

Ok, I'm gonna look for **larapulse.com** at **18.197.136.105**

larapulse.com

18.197.136.105

③

①

«larapulse.com» is at **18.197.136.105**

②

Where is «**larapulse.com**»?

DNS

WEB SERVER

A PIECE OF SOFTWARE CODE THAT LISTENS TO CONNECTIONS AND RESPONDS WITH DOCUMENTS OR OTHER RESOURCES ON THE WORLD WIDE WEB.

APACHE
IIS
NGINX

POPULAR WEB SERVER USED ON THE INTERNET. APACHE CONTROLS THE MAJORITY OF THE MARKET.

HTTP
HTTPS

HYPERTEXT TRANSFER PROTOCOL. A TEXT PROTOCOL CONSISTING OF A SERIES OF COMMANDS USED TO SEND DATA AND REQUEST DOCUMENTS OVER THE WEB.

REQUEST/RESPONSE

THE NAME GIVEN TO THE COMPLETE CYCLE TO FETCH DOCUMENTS AND RESOURCES USING HTTP. THE CLIENT INITIATES THE REQUEST, AND THE SERVER SENDS THE RESPONSE.

FTP/SFTP

FILE TRANSFER PROTOCOL. A METHOD OF TRANSFERRING DATA BETWEEN SERVERS. SFTP IS SECURE-FTP

SECURE SHELL (SSH)

AN ENCRYPTED METHOD OF CONNECTING TO REMOTE COMPUTERS AND EXECUTING COMMANDS.

GIT MERCURIAL SUBVERSION

POPULAR REVISION CONTROL SYSTEMS. THESE SYSTEMS TRACK CHANGES IN FILES ALLOWING EASY RECOVERY OF PREVIOUS VERSIONS.

Classic Web Application Architecture

BASIC BUILDING BLOCKS FOR MOST WEB APPLICATIONS

The old (and current) way

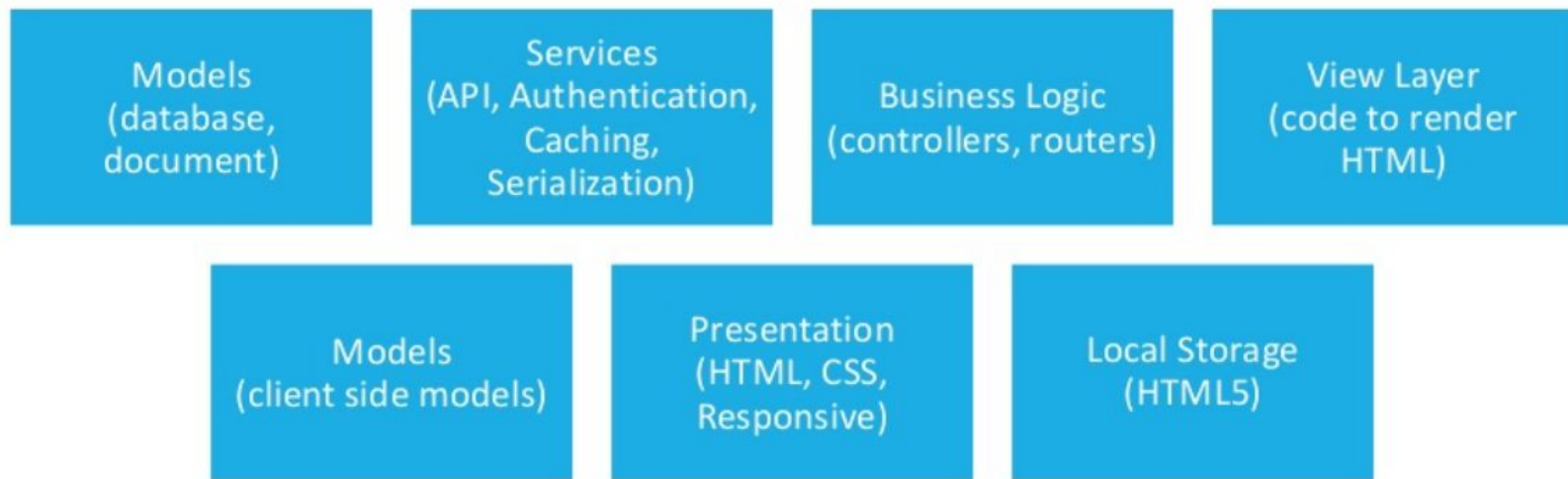|		
DATABASE (the models)	Server Code (controllers)	View (code to render HTML)

Client Side Javascript (jquery components)	HTML Templates; CSS; layouts

Modern Web Application Architecture

ADVANCED APPLICATIONS – FOR ADVANCED DEVICES

Modern Applications (Simplified)

Models (database, document)	**Services** (API, Authentication, Caching, Serialization)	**Business Logic** (controllers, routers)	**View Layer** (code to render HTML)
	Models (client side models)	**Presentation** (HTML, CSS, Responsive)	**Local Storage** (HTML5)

Thanks to

———

1. https://www.slideshare.net/burhankhalid/gust-webdevelopment2015apr
2. https://www.slideshare.net/randyconnolly/web-i-01-introduction-to-web-development