

# Web Application Architecture

CSCI 3300/5300

## Announcements

- Course Evaluations are now open
- May 3 – sprint 3 ends
- May 3 & May 6 – Project presentations
- Project presentation grading criteria is available

## What is a web application?

- Is website a web application?
- Activity: List some popular web sites
- Activity: List some popular web applications

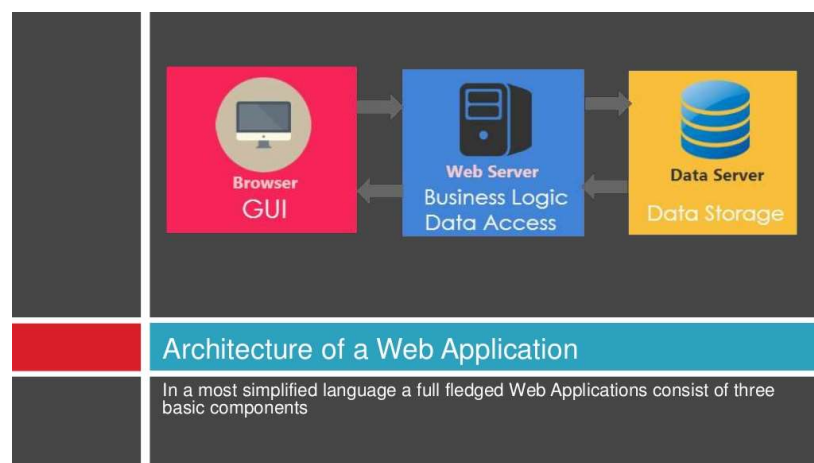
## Activity

- Let's list probable components of a web site

## Activity

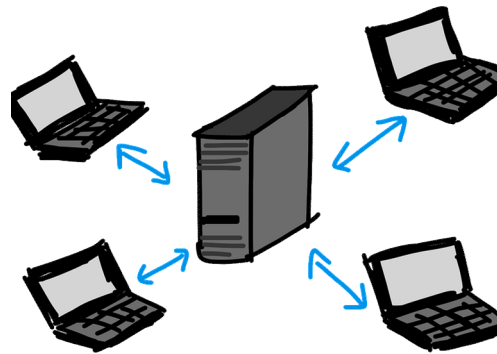
- Let's list probable components of a web app, Facebook for example.

## What is a Web Application



## Client-Server Model

- Server
  - Management and maintenance of Data including
    - User login data
    - Application data
  - Data processing
  - Centralized
  - Access via Login
- Client
  - Provides User Interface
  - Stores some settings
  - Can do some data processing
  - Little or no application data storage



## Client-Server Advantages

- Centralized Data Storage
- No data redundancy (no duplication of data)
- Reduces data dependencies
  - If data is stored on each user's system and each system is different than data depends on how the user system is designed
  - Data can not be shared easily if such dependencies exist

### Classic Example: Early Banking Systems

- **Network:** Local Area Network (LAN) covering local office branch.
- **Server:** Mainframe-like server “in the back” running custom banking system
- **Client:** Windows PC with client interface for each bank teller.
- Data is the same no matter what teller you go to.
- Data is NOT the same if you go to another branch unless servers exchanged some data at night.

### Classic Example: Early Banking Systems

#### **The Obvious Future:**

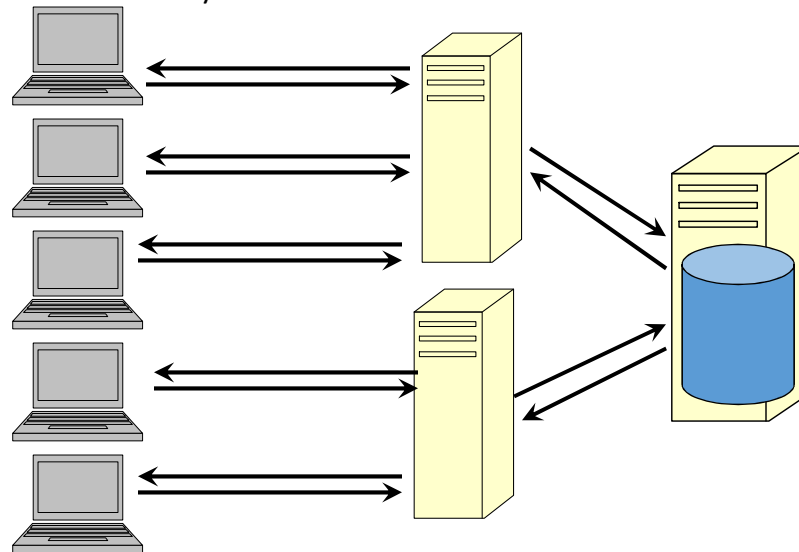
- Change the LAN to a wide area network covering all the branches.
- Get rid of the individual servers at each branch
- Have clients connect to central server where ALL the banking data is stored.

## Classic Example: Early Banking Systems

### The Obvious Problems:

- Large banks could have thousands of tellers connecting to the central server.
- Combining data from all branches requires servers with lots of storage capacity.
- Branch data could be stored in different formats.
- Lack of Standardization.

### 3-Tiered Systems



## 3-Tiered System

- **Database Tier (Database Server)**
  - Data storage and low level data manipulation
- **Server Tier (Application Server)**
  - Manage client connections and data processing
- **Client Tier (Client Software installed locally)**
  - User interface and some data processing

## Advantage of 3-Tier Systems

- Central Database Server accessed by multiple Application Servers
- In turn, each Application Server could independently manage thousands of users
- **Database Server** is specially designed to do its job
  - Database Operations: Update, Insert, Remove, etc.
  - Lots of disk storage and memory needed
- **Application Servers** can be added to support more users or **DIFFERENT APPLICATIONS**
  - Server Operations: Complex application-dependent computations
  - Lots of processor power needed

## "Family Tree" as a Web Application



- application
- data
- database
- driver
- user\_interface

## Many options

- User Interface:
  - Mobile applications:
    - Android
    - IOS
  - Web pages:
    - HTML
    - JavaScript
- Application Server
  - Java
  - Ruby
  - Node.js
  - PHP
  - Python
- Database
  - MySQL
  - DB2
  - Oracle
  - MongoDB

