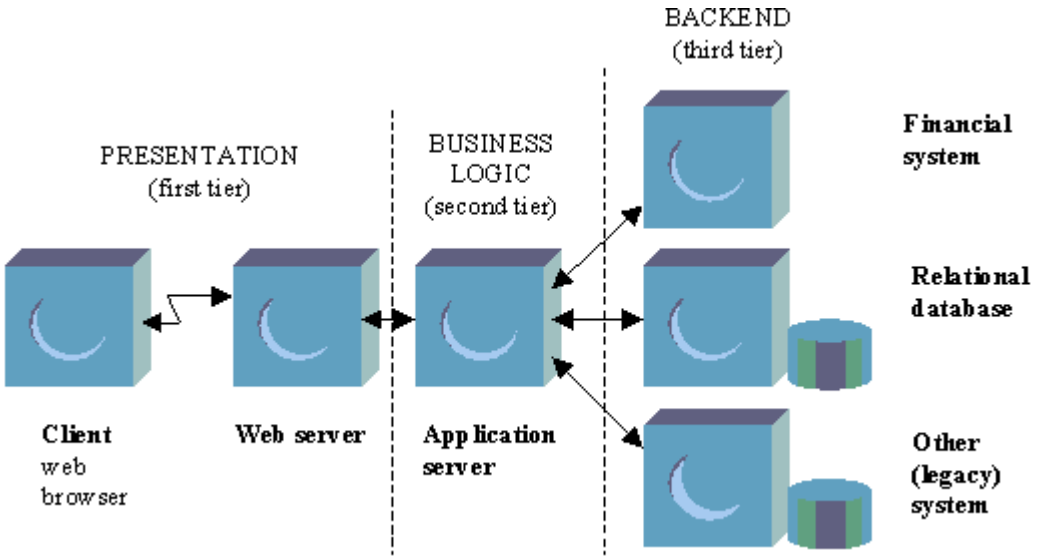
Three Tier Architecture

Team Wildcards

Azizi Penn
Chandini Nagendra
Dane Jew
Steven Mackey

CSC 230, Spring 2018
April 19th, 2018

Architecture Style Purpose



Architecture Style Purpose

- Ensure that critical tasks of a system are separated into modules at different tiers so that any modification or enhancement will not affect the development of the modules at the other tiers
    - The user interface in the “Presentation Layer”, aka. the “first tier” or “Front End”.
    - The data retrieval and internal logic in the “Application Layer”,
        - aka. the “middle tier” or “middle -ware”.
        - Some diagrams call it “Business Logic”.
    - The database and database management system software is housed in the “Data Layer”, aka. the “Back End.”

Applicability

Example: Three Tier System for a website for movie reviews

- Presentation Layer:
  - The User Interface, including form logic and validation.
  - Allows user to create account and enter review data.
  - Housed on a web server.

- Application Layer:
  - Web API, Implemented by a Web service or REST service.
  - Middleware. Contains application logic.
  - Solely responsible to make requests to dbms to create, retrieve, update, and delete data.

- Data Layer:
  - The backend contains the database, dbms, and perhaps a files system.

Strengths/Advantages of Three Tier Architecture

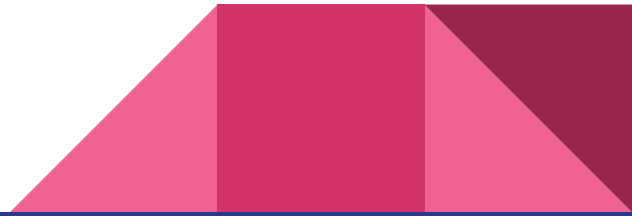
- Lower Coupling
  - Can make changes, updates to one tier with minimal(if any) impact to the other tiers.
  - Especially useful for interfacing with legacy systems and/or multiple backend systems.

- Productivity
  - Software developers can work on separate parts of the application
  - Software developers can work within their expertise.

- Scaleable/Redundancy
  - Greater ability to scale backend servers thus handling larger amounts of traffic.

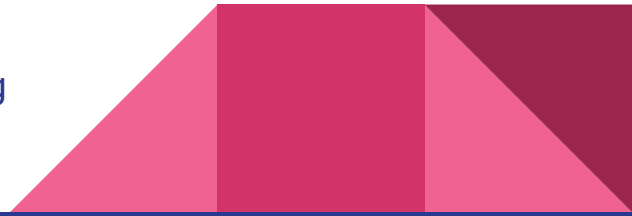
Strengths/Advantages of Three Tier Architecture

- Security
  - Offers greater security of data layer, since client layer does not have direct access to it.
- Maintainability
  - Technology upgrades can be made on one tier at a time.
- Data Hiding
  - Presentation layer does not need to expose all functionality of application layer. Data layer does not need to expose all data to the application layer.



Concerns/Issues of Three Tier Architecture

- Complexity
  - Communication points are doubled when compared to client-server architecture. Need to create interface between client and middle tier, Middle tier and backend.
- Cost
  - Adding additional tier increases the overall cost of the system
- Effort
  - Train existing developers to use different technologies to build 3- different tiers
  - Hire new developers
  - Need more developers to build an additional tier.
  - Need more time to perform integration and end to end testing



Concerns/Issues of Three Tier Architecture

- Reliability
  - Tiers are statistically independent
  - The system functions if all tiers of the system function

  Let E be an event where the system functions as expected
  - Let Ei be an event where the ith tier functions as expected. Where i= 1,2,3
  - P(E) is the probability of the 3-tier system functioning properly
  - Then, P(E) = P(E1)*P(E2)*P(E3)
  - That means the Probability of the system working reliably would be less than the probability of the least reliable component

Ex: Tier 1 = 92%, Tier 2 = 85%, Tier 3 = 95%

System reliability is only 74%

Issue Management

- Cost
  - Leverage Web/cloud based services
- Effort
  - Hiring developers who are flexible/trained to work on different technologies
  - Making use of continuous integration technique
- Reliability
  - Improving the reliability of each tier.

Questions?