Web Development

Full Stack
Architectural Pattern

DEUCENG
Dokuz Eylül University
Dept of Computer Engineering

CME4414 Advances in Web Technologies

Front & Back End

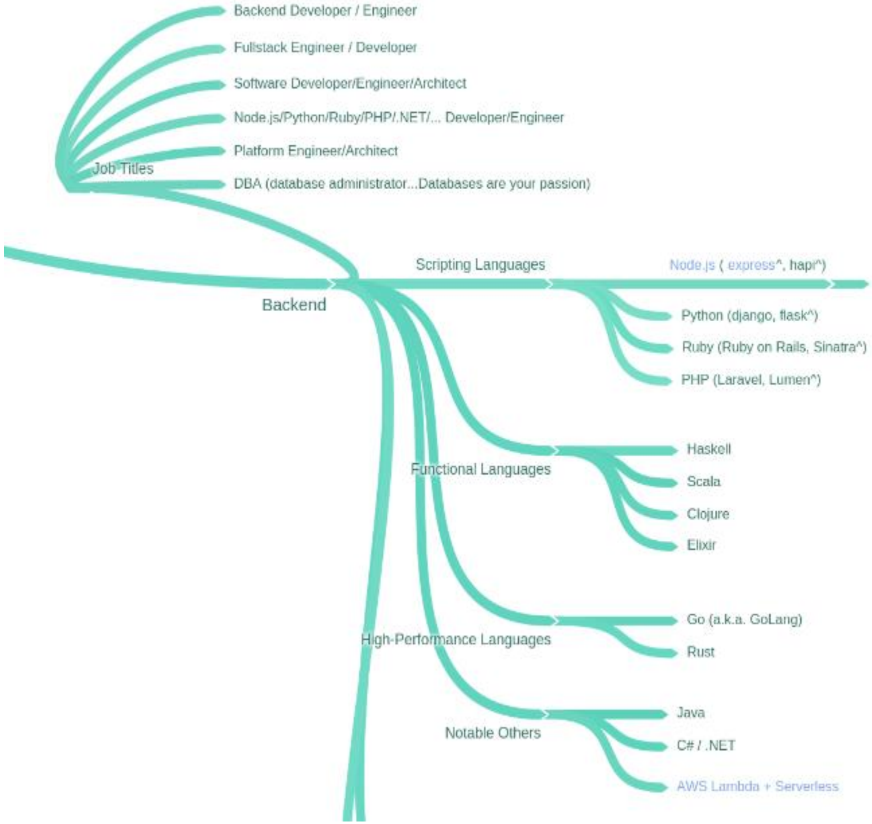
- The separation of concerns between
  - the presentation layer (front end)

    &
  - the data access layer (back end)

- of a piece of software, or the physical infrastructure or hardware

Backend

- Job Titles (Node.js, PHP, .Net,…)
- Languages (C#, .NET)


Job Titles
Backend Developer / Engineer
Fullstack Engineer / Developer
Software Developer/Engineer/Architect
Node.js/Python/Ruby/PHP/.NET/... Developer/Engineer
Platform Engineer/Architect
DBA (database administrator...Databases are your passion)
Backend
Scripting Languages
Node.js (express^, hapi^)
Python (django, flask^)
Ruby (Ruby on Rails, Sinatra^)
PHP (Laravel, Lumen^)
Functional Languages
Haskell
Scala
Clojure
Elixir
High-Performance Languages
Go (a.k.a. GoLang)
Rust
Notable Others
Java
C# / .NET
AWS Lambda + Serverless

Fullstack Developer

Develop both

- **Client** software

  &

- **Server** software

- In addition to mastering HTML and CSS,
- Knows how to
  - Program a **browser** (JavaScript, jQuery, Angular, Vue)
  - Program a **server** (PHP, ASP, Python, Node)
  - Program a **database** (SQL, SQLite, MongoDB)

Full Stack

- LAMP Stack

Linux, Apache, MySQL, PHP

- LEMP Stack

Linux, Nginx, MySQL, PHP

- MEAN Stack

MongoDB, Express, AngularJS, Node.js

- PHP is interchangeable with the languages Python and Perl
- JavaScript can be added

- MVC Pattern
- JSON – Data transfer

Full Stack

Meteor - new web application framework

- Not technically a web stack
- Created to build full-stack applications
- Written in Node.js
- Designed to make the MEAN stack easier
- Exist JavaScript frontend frameworks (React or AngularJS)

Full Stack

Bitnami-hosted Stacks - Cloud-based lib.

- Django Stack

Python, Django, Apache, MySQL
OOP environment

- Ruby on Rails

Ruby, SQLite, PHP

- Java + Spring – The Lightweight Stack

Spring

- Not require programmers to implement any interface
- Helps you connect components together using a system called dependency injection
- Divided into various modules (JDBC, MVC, Core, Data, ORM)

Client Software

- Front End

- HTML
- CSS
- Bootstrap
- JavaScript
- HTML DOM
- JSON
- XML
- jQuery
- Angular
- React
- Backbone.js
- Express.js
- Ember.js
- GraphQL
- ………
- ………

Server Software

- Back End

- PHP
- ASP
- C++
- C#
- Java
- Python
- Node.js
- Ruby
- REST
- GO
- SQL
- MongoDB
- Firebase
- …..
- ….

Back End Focused

- Scripting languages like Node.js, PHP, Python, Ruby, or Perl
- Compiled languages like C#, Java or Go

- Automated testing frameworks for the language being used

- Application Data Access

- Application Business Logic

- Database administration

- Scalability (is the property of a system to handle a growing amount of work by adding resources to the system)

Back End Focused

- High availability
- Security concerns, authentication and authorization
- Software Architecture
- Data transformation
- Backup methods and software

Front End Focused

- Markup and web languages (HTML, CSS, JavaScript, and ancillary libraries commonly used in those languages such as Sass or Jquery)

- Asynchronous request handling and AJAX

- Single-page applications (frameworks React, AngularJS or Vue.js)

- Responsive web design

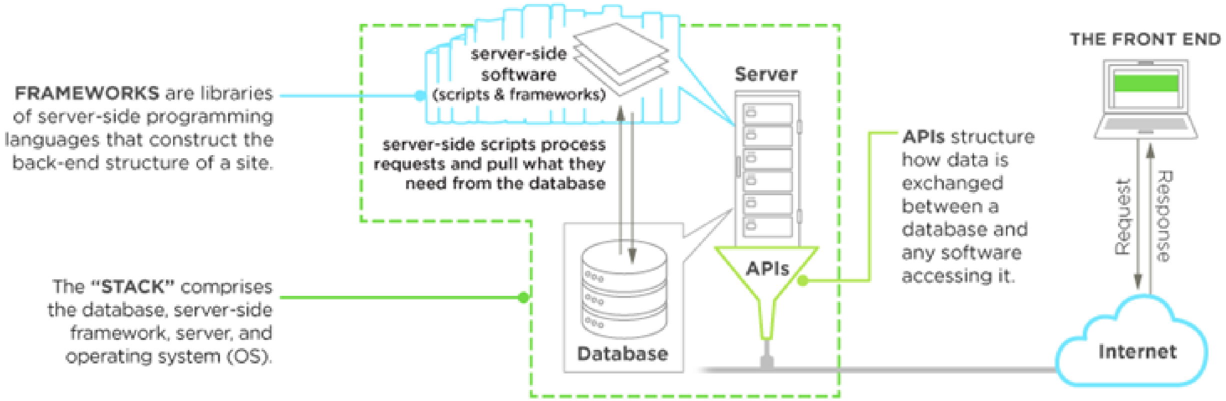
- Web performance (first meaningful paint, time to interactive, 60 FPS animations and interactions, memory usage, etc)

Front End Focused

- Cross-browser compatibility issues and workarounds
- End-to-end testing with a headless browser
- Build automation to transform and bundle JavaScript files, reduce images size, ..(with Webpack, Gulp.js tools)
- Search engine optimization
- Accessibility concerns
- Basic usage of image editing tools such as GIMP or Photoshop
- Visual Interface

Back End Development

Back end Development & Frameworks in server side software



FRAMEWORKS are libraries of server-side programming languages that construct the back-end structure of a site.
server-side software (scripts & frameworks)
server-side scripts process requests and pull what they need from the database
Server
THE FRONT END
APIs structure how data is exchanged between a database and any software accessing it.
Request
Response
The "STACK" comprises the database, server-side framework, server, and operating system (OS).
Database
APIs
Internet

Full Stack Web Developer –Adv.

- Master all the techniques involved in a development project
- Make a prototype very rapidly
- Provide help to all the team members
- Reduce the cost of the project
- Reduce the time used for team communication
- Switch between front and back end development based on requirements
- Better understand all aspects of new (upcoming) technology

Full Stack Web Developer –Disadv.

- The solution choosen can be wrong for the project
- The solution choosen can be dependent on developer skills
- The solution can generate a key person risk
- Being a full stack developer is increasingly complex

Architectural Pattern

- An architectural pattern is
    - General solution

      &
    - Reusable solution

to a commonly occurring problem in software architecture within a given context

- Architectural patterns are similar to **SW design patterns**

  BUT

  have a broader scope

Architectural Pattern

- The architectural patterns address various issues in software engineering,
    - computer hardware performance limitations,
    - high availability
    - minimization of a business risk
- Some architectural patterns implemented within **software frameworks**

Examples of Architectural Pattern

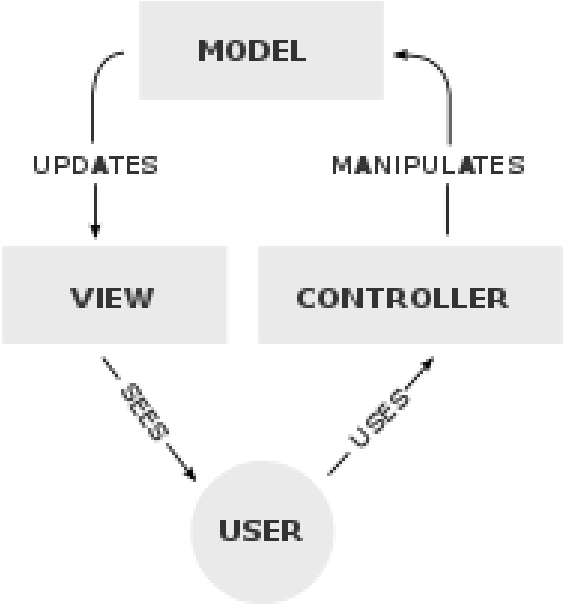
- Blackboard pattern
- Layers (OOD)
- Model–view–controller (Presentation-abstraction-control, Model-view-presenter, & Model-view-viewmodel)
- Entity–component–system
- Multitier architecture (three-tier or n-tier)
- Operational data store (ODS)
- Peer-to-peer
- Pipe and filter architecture
- Service-oriented architecture
- …..

Model View Controller

- An **architectural pattern** commonly used for developing user interfaces that divides an application into *three interconnected parts*

- ***Separate***
    - *internal representations of information from the ways*
    - *information is presented to and accepted from the user*

- *decouples these major components allowing for*
    - *efficient code reuse*
    - *parallel development*

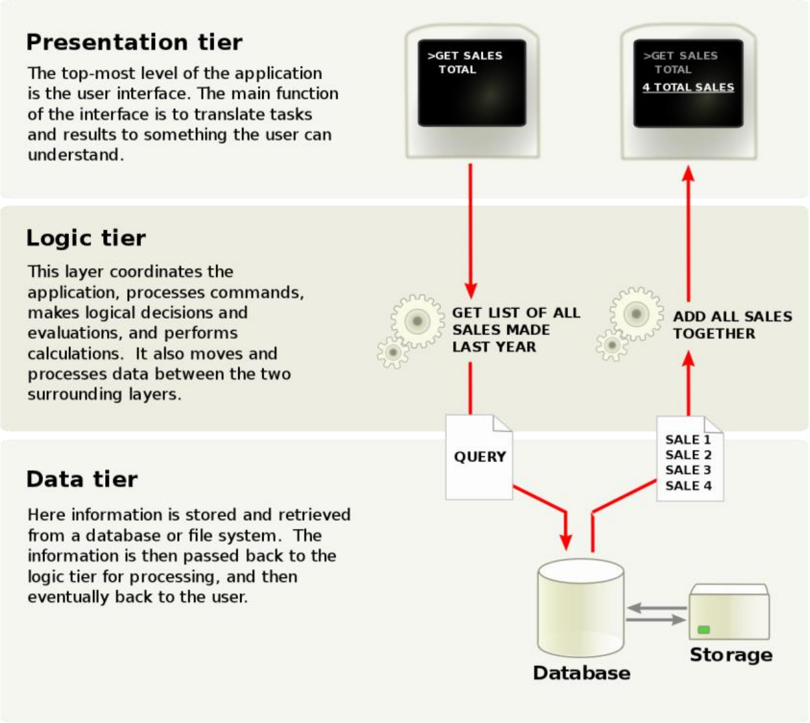


Multitier Architecture

- Multitier architecture (as n-tier architecture, multilayered architecture)

- Client–server architecture in which
  - presentation,
  - application processing,
  - data management functions

are physically separated



Presentation tier

The top-most level of the application is the user interface. The main function of the interface is to translate tasks and results to something the user can understand.

Logic tier

This layer coordinates the application, processes commands, makes logical decisions and evaluations, and performs calculations. It also moves and processes data between the two surrounding layers.

Data tier

Here information is stored and retrieved from a database or file system. The information is then passed back to the logic tier for processing, and then eventually back to the user.

Multitier Architecture

- provides a model by which developers can create
    - Flexible
    - Reusable applications

- n a logical multilayered architecture for an information system with an object-oriented design
    - Presentation layer
    - Application layer
    - Businesss layer
    - Data Access layer

ASP.Net

- .NET is a developer platform made up of tools, programming languages, and libraries for building many different types of applications

- ASP.NET extends the .NET developer platform with tools and libraries specifically for building **web apps**

- .NET Core is a **cross-platform** version of .NET for building websites, services, and console apps

References

- The DCI Architecture: A New Vision of Object-Oriented Programming – Trygve Reenskaug and James Coplien – March 20, 2009
- R. N. Taylor, N. Medvidović and E. M. Dashofy, Software architecture: Foundations, Theory and Practice. Wiley, 2009
- https://en.wikipedia.org/wiki/
- www.w3schools.com