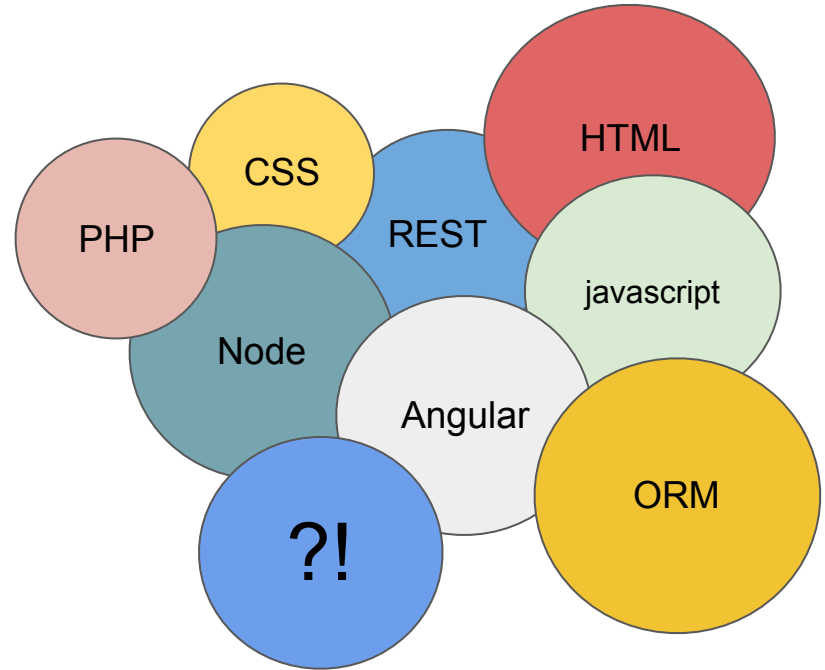
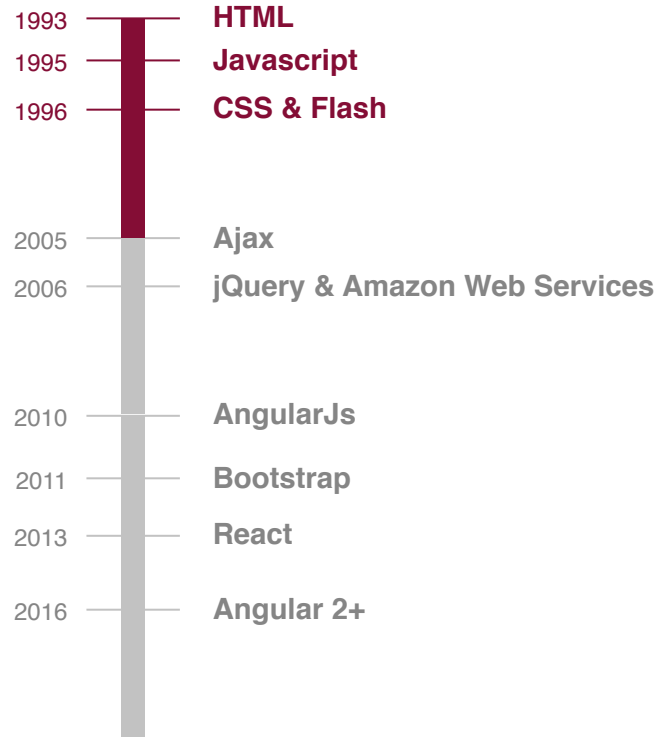


Practical web development basics



Prepared by Nitish Patkar

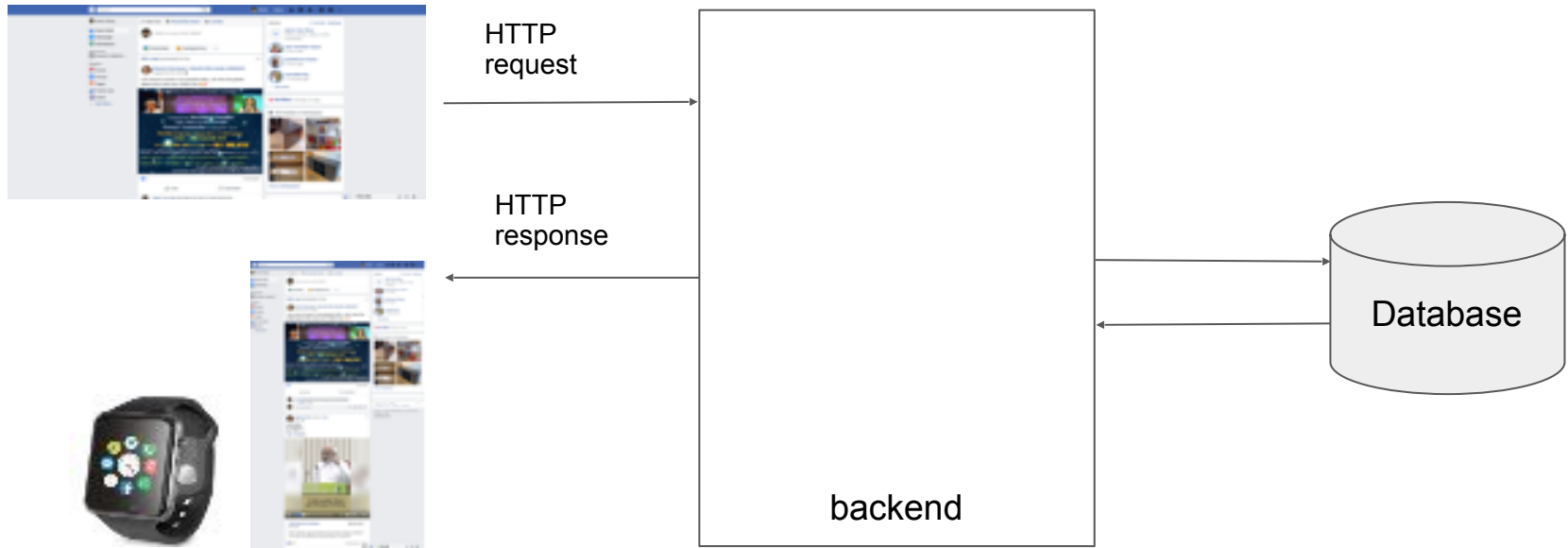
Major milestones in web development



As a web developer you cannot live without..



Simplified view on today's web development



Various frontends

Frontend: key responsibilities and main concepts to get familiar with



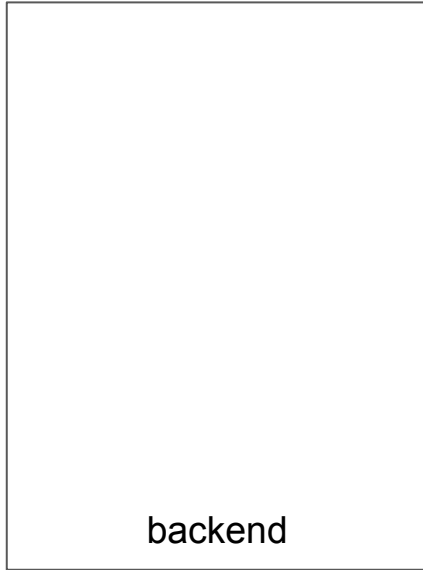
Various frontends

- It is what your users see- can be accessed on mobile as an app, or as a mobile website, or on laptops etc.
- Accepts request from users and forwards it to the backend, subsequently receives response from backend and shows it to the user

Main concepts to get familiar with-

- HTML, CSS, Javascript, Typescript
- HTTP requests
- Ajax (or asynchronous calls to backend)
- Javascript frameworks and libraries (e.g. Angular, React, Vue.js)
- Styling frameworks (e.g. Bootstrap, Material)
- Package manager (e.g. NPM)
- Single Page Applications (SPAs)
- Developer tool in browser for debugging
- Responsive design
- Browser compatibility
- ...

Backend: key responsibilities and main concepts to get familiar with

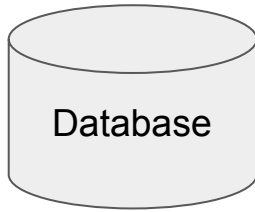


- Handles business logic for your application- you write a single backend application which is typically consumed by various frontends
- Coordinates with frontend and database

Main concepts to get familiar with-

- REST API
- JSON data format
- Programming languages (e.g. PHP, Python, Java)
- Backend frameworks (e.g. Django Rest, .NET core, Ruby on Rails)
- Object Relational Mapping (ORM)
- Request routing
- Security concerns
- Architecture (e.g. MVC, MVVM)
- Session management

Database: key responsibilities and main concepts to get familiar with



- Stores user and application data
- With the availability of ORMs, developers deal less and less with databases directly, but it is still good to know

Main concepts to get familiar with-

- Types of databases (e.g. Relational, non-relational)
- Database backups
- Database query languages (e.g. SQL)
- Scripting languages (e.g. PL/SQL)

Frameworks

- One can do web development without frameworks, but using them simplifies many routine tasks and speeds up the development, also makes sure applications are scalable
- Choosing a correct framework depends on multiple factors such as programming language preference, framework documentation, community support etc.
- There are different frameworks for frontend development, backend development, and also for styling

Frontend (javascript) frameworks

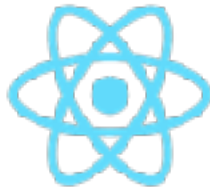
- Enforces developers to write HTML, CSS and javascript in a certain way to make the application easier to develop and maintain
- Easy handling of Ajax calls, HTTP requests
- Most popular frontend frameworks and libraries:



Angular



Vue.js



React



Ember.js

Styling (CSS) frameworks

- Provides ready to use css classes, components and grid system based on proven design standards, so that developers do not have to write css from scratch
- Most popular css frameworks:



Bootstrap



Semantic UI



Materialize



Material UI

Backend frameworks

- They provide tools and libraries that simplify common web development tasks, including routing URLs to appropriate handlers, interacting with databases through ORM, supporting sessions and user authorization, formatting output (e.g. HTML, JSON, XML), and improving security against web attacks.
- Most popular backend frameworks:



Ruby on Rails



Laravel



Flask

Languages: HTML, CSS, Javascript
Frameworks: Angular, React, Vue.js
Styling: bootstrap, Material UI, Saas
Concepts: HTTP requests, responsive design, asynchronous calls, SPAs, NPM

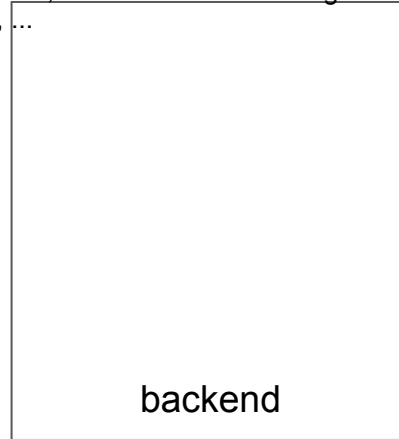
Languages: HTML, CSS, Javascript
Frameworks: Angular, React, Vue.js
Styling: bootstrap, Material UI, Saas
Concepts: HTTP requests, responsive design, asynchronous calls, SPAs, NPM



Various frontends

Languages: Python, Java, javascript,...
Frameworks: Django, .NET core, Laravel,...
Concepts: HTTP requests routing, REST API, ORM, token & session management, CSRF, ...

Languages: Python, Java, javascript,...
Frameworks: Django, .NET core, Laravel,...
Concepts: HTTP requests routing, REST API, ORM, token & session management, CSRF, ...



Types: Relational, NoSQL
Examples: MySQL, PostgreSQL, MongoDB,..

Types: Relational, NoSQL
Examples: MySQL, PostgreSQL, MongoDB,..



Project scaffolding

- Helps to quickly set up a project skeleton (hopefully) with best practices
- One can scaffold a project by using CLI tools such as angular cli for angular projects, or by cloning a repository from GitHub, gitlab etc.
- Best known scaffolding tools:



Yeoman



Slush

Project management tools

- Helps to organise development work within teams, manage user stories, deadlines, generate project analytics and many more tasks
- Best known project management tools:



Communication tools

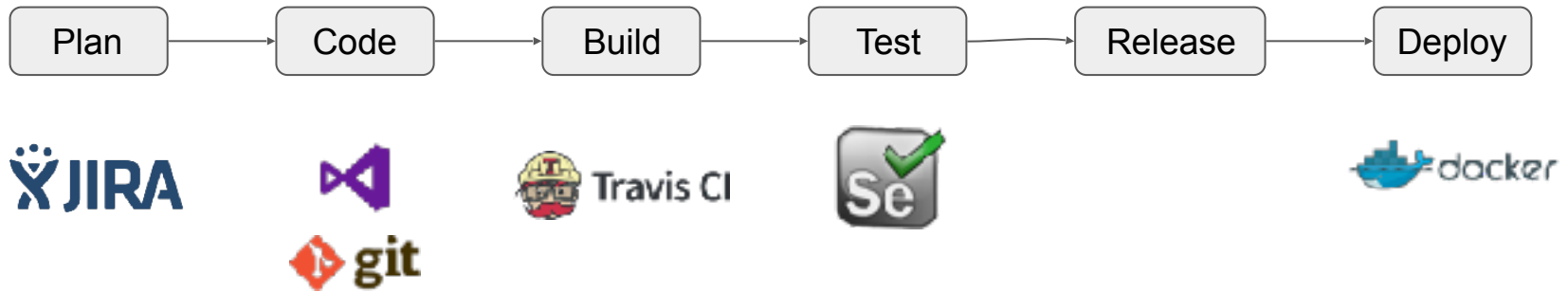
- Helps to communicate with other team members through dedicated channels, direct messages and rich content such as video, images, Gif etc.
- Provide search functionality and integrations to other applications such as confluence, gitlab etc.
- Best known communication tool:



Latest buzzwords in web development

- Agile
- CI/CD pipeline
- DevOps

Continuous Integration (CI)/Continuous Delivery (CD) pipeline and toolchain



Other end to end solutions



