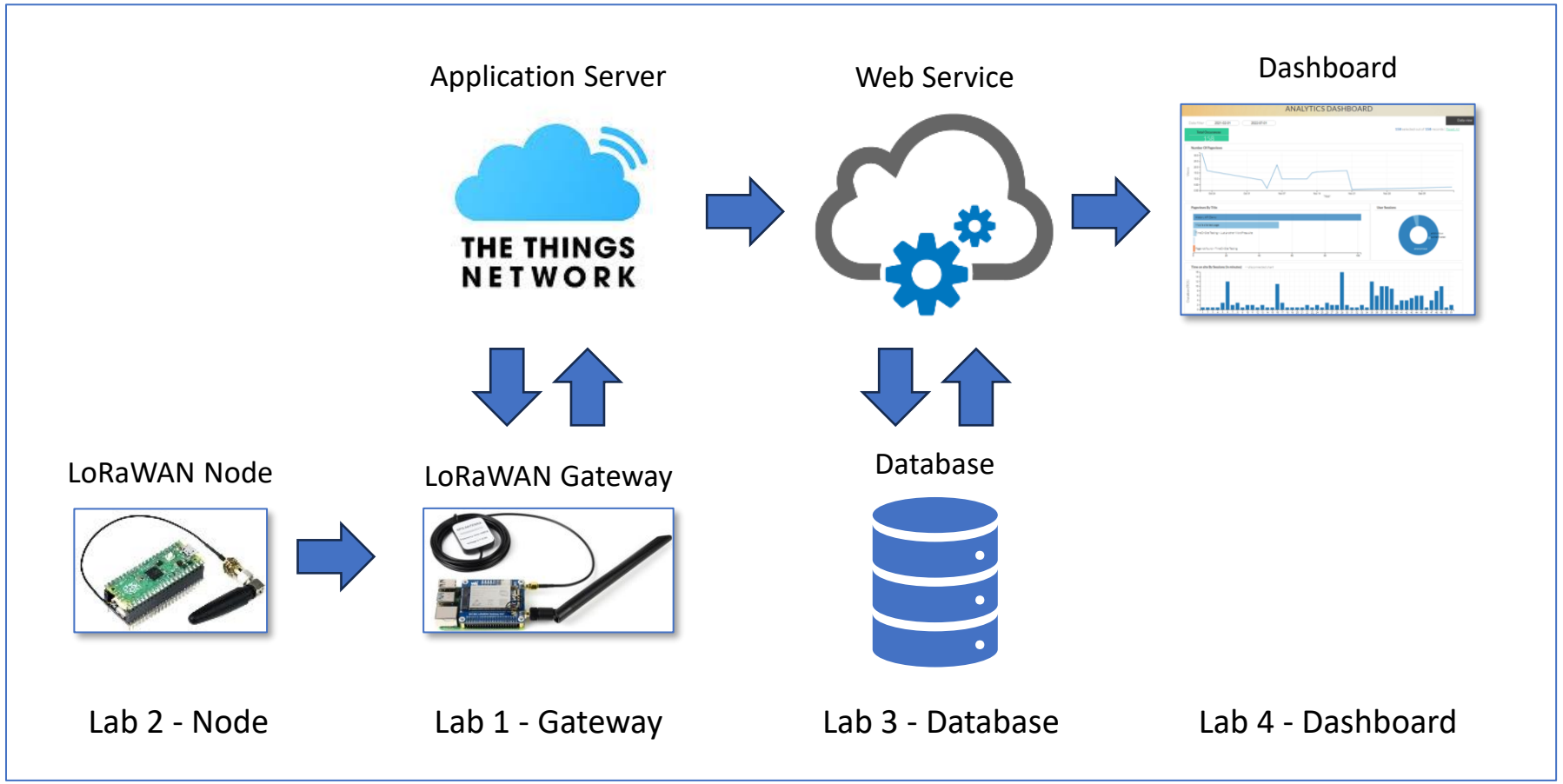
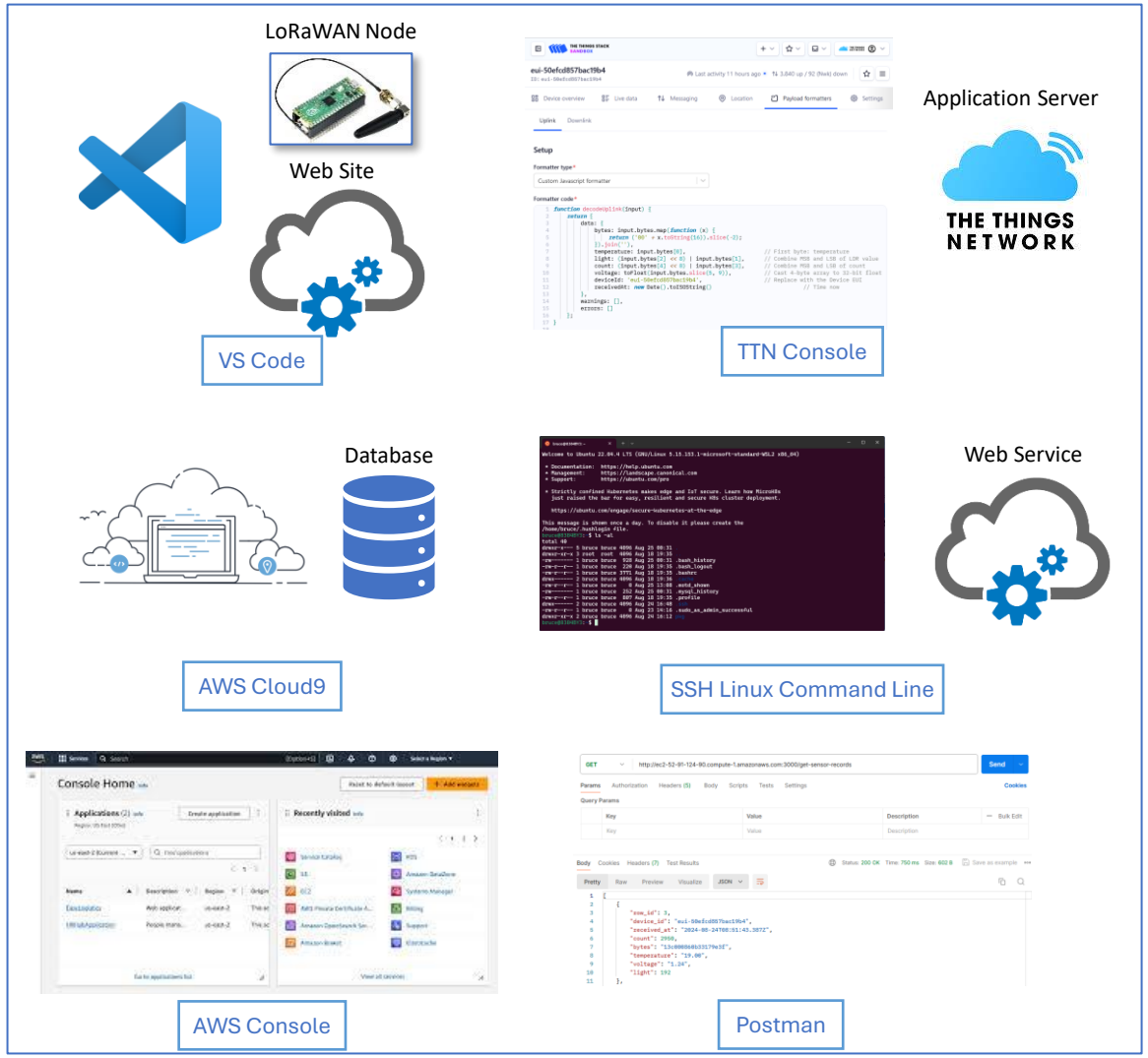
EE4500 – Web Development

JAMES COOK UNIVERSITY
AUSTRALIA

EE4500 Labs – The big picture



EE4500 Labs – Activities



Schedule

Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7
Introduction	AWS Introduction	AWS Global Infrastructure Overview	AWS Compute	AWS Storage	AWS Certification Walhthrough	
RESTful APIs	Cloud concepts overview	AWS Networking and Connectivity		AWS Databases	Subject Wrap	
	LoRaWAN & TTN		Web development			
Business Models (BMC/VP)	Value Proposition Design	Project Review & Feedback	3D Printing	Project Review	Architecture Quiz	
Setup LoRaWAN Gateway	LoRaWAN Node	IoT Database	IoT Dashboard			Presentation

Legend
In-person lectorials
Activities
On-line lectures
Labs

Tools, platforms and frameworks

- Operating system
- Hardware
- Languages
- Web frameworks
- Cloud platforms
- Pipelines

Platform

- **Front-end**:
  - Web browsers (Chrome, Firefox, Safari, Edge)
- **Back-end**:
  - Web servers (Apache, Nginx, IIS)
  - Cloud platforms (AWS, Azure, GCP)
- **Mobile**:
  - Web, native, or hybrid apps

Programming Languages

- **Front-end**: HTML, CSS, JavaScript, TypeScript
- **Back-end**: Node.js, Python, Ruby, PHP, Java, C#, Go

Frameworks & Libraries

- **Front-end**: React, Vue.js, Angular, Svelte
- **Back-end**: Express.js, Django, Flask, Spring, Ruby on Rails

Tools

- **Version Control**
  - Git (GitHub, GitLab, Bitbucket)
- **IDEs/Editors**:
  - Visual Studio Code, WebStorm, Sublime Text, Atom

Database Management

- **SQL**: MySQL, PostgreSQL, SQLite, MSSQL
- **NoSQL**: MongoDB, Firebase, CouchDB

APIs & Web Services

- REST APIs
- GraphQL
- SOAP

Hosting & Deployment

- **Web hosting**: Shared hosting, VPS, dedicated servers
- **Cloud**: AWS, Azure, Google Cloud, Heroku, Netlify

Security & Authentication

- HTTPS
- SSL/TLS
- Oauth
- JWT (JSON Web Token)
- SAML

Performance Optimization

- CDN (Content Delivery Network): Cloudflare, Akamai
- Compression, caching, lazy loading

Testing & Debugging

- **Unit testing**: Jest, Mocha
- **End-to-end testing**: Selenium, Cypress
- **Debugging**: Chrome DevTools, Firebug

Responsive Design

- **Frameworks**: Bootstrap, Foundation, Tailwind CSS
- **Media queries & mobile-first design**: CSS Flexbox, Grid

SEO

- Search Engine Optimization
  - Keyword optimization
  - meta tags
  - Sitemaps
  - Google Analytics
  - schema markup

Overview of Platforms

Feature	Windows	Linux	macOS
Popularity	Widely used, especially in corporate environments	Preferred by developers for flexibility and open-source tools	Common in design and development communities
Command Line	PowerShell, CMD (native), WSL (for Linux compatibility)	Bash/Zsh, highly customizable	Terminal with Unix-based commands
Cost	Licensed (paid)	Free and open-source (various distros)	Paid with Apple hardware
Security	Vulnerable to malware	Highly secure, but depends on user config	Strong security built-in

Development Environment & Tools

Aspect	Windows	Linux	macOS
Tool Availability	Visual Studio, VS Code, .NET, Node.js	Comprehensive toolset (open-source), supports Node.js, Python, etc.	Xcode, VS Code, Unix utilities
Web Server Support	IIS, XAMPP	Nginx, Apache, LAMP stack	Apache (native), Nginx
Package Management	Chocolatey, Winget	APT, Yum, Snap, Pacman (depending on distro)	Homebrew
Virtualization & Docker	Good, with WSL2 support	Seamless, lightweight integration	Efficient with native Unix support

Performance & Compatibility

Factor	Windows	Linux	macOS
Performance	Slower with resource-intensive apps like Docker without WSL2	Optimized for performance, highly efficient	Well-optimized for development tasks
Compatibility	Extensive compatibility with enterprise software	Works best with open-source and Unix-based software	Compatible with Unix-based systems, but some proprietary tools
Community Support	Large community, lots of tutorials	Huge open-source developer community	Strong developer community, but focused on macOS users

Platform summaries

- Linux
  - UI is for enthusiasts only
  - Command line is superb
- Windows
  - UI is market leading
  - Command line is inconsistent and weak (Powershell…)
  - WSL
- Mac
  - Acquired taste
  - Popular for running Linux

Platform outcomes

- WSL
- Linux on Windows
- Best of both worlds
- Try it

Overview of Front-End Languages

Language	HTML	CSS	JavaScript	TypeScript	WebAssembly
Purpose	Structure and content of web pages	Styling and layout of web pages	Interactivity and dynamic behavior	Superset of JavaScript with static typing	High-performance code for web apps
Type	Markup language	Style sheet language	Scripting language	Typed programming language	Binary instruction format
First Released	1993	1996	1995	2012	2017
Learning Curve	Easy	Easy to excruciating	Moderate to hard	Harder (requires JavaScript knowledge)	Hard (requires knowledge of low-level languages)

Features & Functionality

Language	HTML	CSS	JavaScript	TypeScript	WebAssembly
Core Functionality	Provides page structure (headings, paragraphs, links, etc.)	Adds styling (colors, fonts, layout) to HTML elements	Adds dynamic content, interactivity, animations	Adds type safety, interfaces, and better tooling to JavaScript	Runs code from languages like C, C++, and Rust in browsers
Frameworks/ Libraries	(JSX)	Bootstrap, Tailwind CSS	React, Vue, Angular	Works with JS frameworks	Works with JS, Rust, C, C++
Rendering Control	Controls DOM structure	Controls look and feel of web elements	Controls behavior, DOM manipulation, event handling	Same as JavaScript with better tooling support	Handles performance-intensive tasks
Reusability	Very reusable (templates, semantic HTML)	Reusable with classes and CSS-in-JS	Reusable with modules, functions	Stronger type system improves reusability	Modularized and compiled components

Performance & Use Cases

Factor	HTML	CSS	JavaScript	TypeScript	WebAssembly
Performance	Light and fast	Very efficient	Moderate, can be slow with large scripts	Comparable to JavaScript with better debugging	Extremely fast (runs at near-native speed)
Use Cases	Essential for every web page	Essential for styling, responsive design	Interactive websites, single-page apps, client-side scripting	Large-scale applications, where type safety and refactoring are critical	High-performance web apps, games, complex algorithms
Browser Support	Supported by all browsers			Supported by all modern browsers	
Community & Ecosystem	Large, mature ecosystem		Largest ecosystem with the most tools and libraries	Growing rapidly with a focus on enterprise apps	Growing, but smaller compared to JS

Outcome

- HTML
  - Either use it directly or generate it
- CSS
  - Required, directly or generated
- JavaScript or TypeScript
  - Very similar. No realistic alternatives at this time.

Integrated Development Environments

- Emacs
- Vim
- VS Code
- Web Storm
- Sublime Text
- Atom
- Brackets

IDE Comparison for Web Development

IDE/Editor	Visual Studio Code	WebStorm	Sublime Text	Atom	Brackets
Type	Code editor with extensions	Full-featured IDE	Lightweight code editor	Code editor	Code editor
Cost	Free	Paid (free evaluation)	Paid (free evaluation)	Free	Free
Performance	Fast, customizable	Heavy, but feature-rich	Extremely fast	Slower with large projects	Lightweight, focused on web
Language Support	Broad via extensions	Excellent JS/TS, HTML/CSS	Multi-language, plugin-based	Supports many languages	Specializes in HTML/CSS/JS
Best For	General web development	Professional JavaScript/TypeScript	Minimalist workflows	General purpose, declining	Front-end web development

Overview of Frontend Frameworks

Feature	React	Vue.js	Angular	Svelte
Released	2013 by Facebook	2014 by Evan You	2016 (Angular 2) by Google	2016 by Rich Harris
Architecture	Library (UI-focused)	Framework (UI-focused)	Full-fledged framework	Compiler (no runtime needed)
Learning Curve	Moderate	Easy to moderate	Steep	Easy to moderate
Type	Component-based	Component-based	Component-based + MVC	Component-based
Popularity	Extremely popular	Growing steadily	Popular in enterprise apps	Gaining traction rapidly

Performance and Flexibility

Factor	React	Vue.js	Angular	Svelte
Virtual DOM	Yes	Yes	Yes	No, uses compiled code
Rendering Speed	Fast with reconciliation	Fast, lightweight	Relatively slower	Extremely fast (no virtual DOM)
Flexibility	Highly flexible, only handles UI, integrates well with other libraries	Flexible, with a simple core and optional features	Less flexible, tightly integrated features	Very flexible, reactive framework without boilerplate
State Management	Requires additional libraries like Redux	Vuex for state management	Built-in services and RxJS	Built-in reactivity without external libraries

Ecosystem and Tooling

Aspect	React	Vue.js	Angular	Svelte
Tooling & CLI	Create React App, Next.js, Gatsby	Vue CLI, Nuxt.js	Angular CLI, extensive tooling	SvelteKit for full-stack apps
Community & Ecosystem	Largest ecosystem, numerous libraries and tools	Growing ecosystem, fewer libraries but strong support	Strong ecosystem, but with more overhead	Smaller but growing, fewer libraries compared to others
Use Cases	Suitable for all types of applications, especially SPAs	Great for small to medium-scale apps	Ideal for large-scale applications	Best for smaller apps or when speed matters
Backward Compatibility	Strong with gradual updates	Excellent with minor breaking changes	Significant breaking changes in updates	Backward compatibility not yet a major concern

Framework outcome

- Vue is preferred by many
  - Much smaller ecosystem
- Angular is very hard work
- Svelte is up and coming
  - Try it if you have spare time
- React has a huge ecosystem
  - We are using React

Major Cloud Platforms Comparison

Platform	AWS (Amazon Web Services)	Microsoft Azure	Google Cloud Platform (GCP)
Founded	2006	2010	2008
Market Share	Largest (~33%)	Second-largest (~22%)	Third-largest (~10%)
Strengths	Vast service offerings, most mature, global reach	Strong hybrid cloud capabilities, seamless with Microsoft products	Excellent in data analytics, AI, and machine learning services
Pricing	Complex, pay-as-you-go, per-second billing	Pay-as-you-go, enterprise discounts, flexible	Competitive, pay-per-use, user-friendly pricing
Compute Services	EC2, Lambda	Virtual Machines, Azure Functions	Compute Engine, Cloud Functions
Storage	S3, EFS, Glacier	Blob Storage, Azure Files	Cloud Storage, Persistent Disks
AI/ML Tools	SageMaker, Rekognition	Azure AI, Cognitive Services	TensorFlow, Vertex AI

Cloud services outcome

- AWS has a bigger support network
- But UniSuper tells us to use more than one

Take-away

- Web development is complex
- Multi-faceted
- More choices than in most software development fields
- You will need to know many different tools, frameworks & technologies
- It is impossible to be fully up-to-date

Next – 3D Printing IoT Dashboard