



1003 SUMMARY



{2017}

Web applications:

Web pages are created using:

- HTML
- CSS
- JavaScript
- Media

Document Object Model (DOM): an in-memory representation of the document

Web applications: dynamic web pages that allow a user to interact with the page

HyperText Markup Language (HTML): content & structure

- Structure: defines the content & specify semantics

Cascading Style Sheets (CSS): list of a number or style rules

- A selector is used to select elements
- One or more declarations are used to style the pairs

JavaScript: a client-side programming language

- Access to DOM with libraries of code called JavaScript frameworks
- Interactions
- Data

Object Oriented Programming

Classes: reusable objects containing data items (attributes) and methods that operate on this data (operations)

- Access: methods
 - Multiple inheritance: methods
 - Inheritance: defining that a class inherits attributes and methods from a particular base class using the extends keyword
- Constructor or parent class

Version Control

- A piece of software is tracked in multiple versions of files

Regression; but you can revert to previous changes, but not previous versions

Repository; type of server, store master code

Workspace; local

Clone; clone of

Staging; making ready to commit

Commit; take snapshots of changes and store on the local

Push; Take changes from local repository & add to remote repository

Pull; Take new changes from remote repository & add to local

Merge; Pulling from remote repository, merge remote & local

Diffs; human readable showing line by line changes

Code documents

- Inline comments
- Header comments
- Replace named variables
- Constants
 - (var)
 -

Storing Data

Cookies; data stored on your computer when you visit a web site and sent back to browser

Local storage; data stored on your computer per web domain

Object data:

Serialisation / Deserialisation; converting object data to a string format stored or transmitted as text

JSON (JavaScript Object Notation)

- Specific syntax for object
- Does not support properties
- Properties in quotes

Software development methodologies:

Agile development

Client is involved in the development process to direct the development

1. Design (test) something
 2. Development & consultation with client
 3. Release
- Use of sprints (short – done)

Waterfall model

Requirements > Design > Development > Verification > Maintenance

- Use of milestones

Project management (PMP);

Project information - aims & scope

Info on personnel - on the project, their roles

Decisions on

Communication - meetings, written reports

Quality Management - assurance

Risk Management - risk & contingency plan

Requirements

- Software engineer involved in gathering requirements - reasonable expectations

Requirements - government legislation, privacy

Requirements

User stories - list of actions a user would want

Individual

Needs

Value

Estimate

Small

Test

Software design:

Class diagrams; Visual representation of the structure of the system

- Standardised part of UML (Unified Modelling Language)
- Often lists; Class name; attributes & methods
- Multiplicities; How many objects of each class are involved
- Cohesion; how conceptually related functionality is. (want ↑ cohesion)
- Coupling; dependencies between classes. (want ↓ coupling)

Generalisation; when one class offers a support f functionality to another

- Implemented via inheritance

Conceptual class diagrams; Very-informal, model problem domains

Activity diagrams; used to document workflows and even overall system behaviour, flow charts for software

Prototyping and UI design; user interface & user experience (ux)

- Plan layout and behaviour

Wireframing; sketches of user interface, showing mock data

- Skeletal view of the app

Storyboards; sequence of user interface steps and connecting transitions used for simulating interaction

Sources of software requirements

- Client desires
- Government regulation
- Industry standards
- Hardware or software environment
- Customer / market research

Software verification

Testing the code or software to find errors

Static testing; without running the code

- Code review; looking through by eye
- Walkthroughs; flowing a process & looking at problem cases
- Static analysis;
 - Involves automated analysis of the code

Dynamic testing

Approaches to testing;

1. Black box testing
 - Testing code while knowing what it is supposed to do, but not how it does it
 - Test created from software requirements
 - Has concept of 'oracles'
 - Expectation of the codes output
2. White box testing
 - Code based approach, tests are written knowing how the code works
 - Based on the structure/design of the code

Levels of testing;

- Unit testing;
 - Test small portions, such as a function, class or method
- Integration testing
 - Examine behaviour & communication between components
 - Carried out after unit testing
- System testing;
 - Testing the complete application
 - Test created from 'user stories' or 'activity diagrams'

Coverage testing;

- Program that executes all available test cases
- Counts how many times each line is executed
- Want 100% coverage
- If unrun code, may be an issue

Regression testing;

- Runs every time new change to code is committed, tests doesn't indirectly affect other areas of the code

Internet architecture & web services

Network protocols; TCP / IP

- Internet protocol (IP);
 - Delivery of messages between devices
 - IPv4 (most common)
 - 4 bytes of information – 4 numbers between 0 – 255
 - IPv6
 - 7.9×10^{28} times more addresses than IPv4
 - 8, colon separated groups of four hexadecimal digits

Network address translation (NAT)

- Lets device share a single IP address, then distributes messages accordingly

Domain

- Domain names provide more human names

Web services; Network communication and AJAX

- Loading data into the current page
- AJAX – Asynchronous JavaScript and XML
 - Enable dynamic content

JSONP;

- JSON with Padding (padded with callback function)
- Convenient way of making AJAX requests

URL encoding;

- Removes special characters, so string can be used as a URL

Algorithms & Efficiency

Algorithm; series of instructions to solve a problem or produce a particular outcome

Code complexity;

- Complexity analysis; how optimal algorithms are
 - Time taken
 - Space
 - Resources required
 - Expressed as 'big O notation', $O(\log(n))$

Selecting an algorithm;

- Speed
- Quality of results
- Types of operations performed
- Storage requirements
- Readability & understandability
- Stable results

Mobile Hardware Architecture

Data representation; binary

- Ones & zeros
- Bit – binary digit
 - grouped in multiples of 8, making a byte
- Binary arithmetic;
 - $00101010 = (2^5 + 2^3 + 2^1) = 42$
 - Addition; add 1 & 0 - two 1's carry the 1
 - Subtraction; flip bit + 1 (two's complement) & add

High level con

Processor; the brains, responsible for executing instructions

- Speed; clock cycles per second
- Cache; small local copy of recently accessed information
- Cores; multiple cores allow computer to execute tasks simultaneously

Storage & memory;

- Registers; fastest, on the processor, very few
 - Data being used by CPU
- Cache; very fast, on processor, recent items from RAM
- Random Access Memory (RAM); fast, but temporary
- Hard Disk / Flash memory; slower, permanent storage
 - Flash limited to number of writes

Peripherals;

- Radios; Bluetooth, Wi-Fi, cellular, GPS
- Input / output devices; display, keyboard, mouse

From code to execution

Machine code; code the CPU understands, tedious to write

Assembly language; human readable low-level machine code, tedious

Interpreted languages; instructions interpreted & actions performed by another piece of software

- Run slower than compiled languages

User experience; HCI (Human-Computer Interaction)

Mobile UI vs Desktop UI;

- Brief windows of activity
- Small screen design considerations
- Touch based input methods

Platform conventions

- Design guidelines-advising how to present information
- Skeuomorphism; practice of engineering software to look like a physical device or object

Security &
Strong password

- Long – contains 8 or more characters
- No dictionary words
- Mix of special characters
- Isn't reused for multiple accounts

Dictionary attacks; attempting to log in with dictionary words

Brute force attacks; every possible combination

Rainbow Table attack; find corresponding password for a known hash by looking it up in a table of hashes for all common & short password combinations

Securely storing passwords;

- Worst case, plain text on server
- Cryptographic hash; one-way function
 - Stored on server
 - Rainbow Tables; tables containing pre-calculated hashes for common passwords
- Salting; technique used with hashes counteracting rainbow tables
 - Random string added to password before it is hashed

Social engineering; deceiving & manipulating people to gain information

Two-factor identification; (2FA)

- Need something secret and something possessed by the user

Secure storage & transmission;

- Encryption;
- HTTPS; encrypted between the web server & web browser
- Certificates; cryptographic method of verifying a web server

Heartbleed; security concern in 2014

- Returned extra information if string was shorter than expected

Sandboxing; limiting what applications can access