Computer Systems and Software

Hardware: Motherboard, CPU, RAM, Storage, Peripherals

Software: Firmware, System, Server-Side, Applications

SoftUni Team

Technical Trainers

Software University

SoftUni

Software University

https://about.softuni.bg

Have a Question?

sli.do

#qa-fund

Table of Contents

Computer Systems

Components and Functionality

What is a Computer System?

- **Computer system**: integrated bundle of hardware and software components, e. g. smartphone, POS terminal, laptop
  - Enables **efficient data input, processing,** and **output**
  - Comprises **interconnected software** and **hardware components**
  - **Human-computer interaction** for the end-users / APIs for **machine-to-machine interaction**
- **Key elements**:
  - **Hardware**: RAM, input/output devices, storage devices, CPU
  - **Software**: operating systems, drivers, apps, games

Evolution

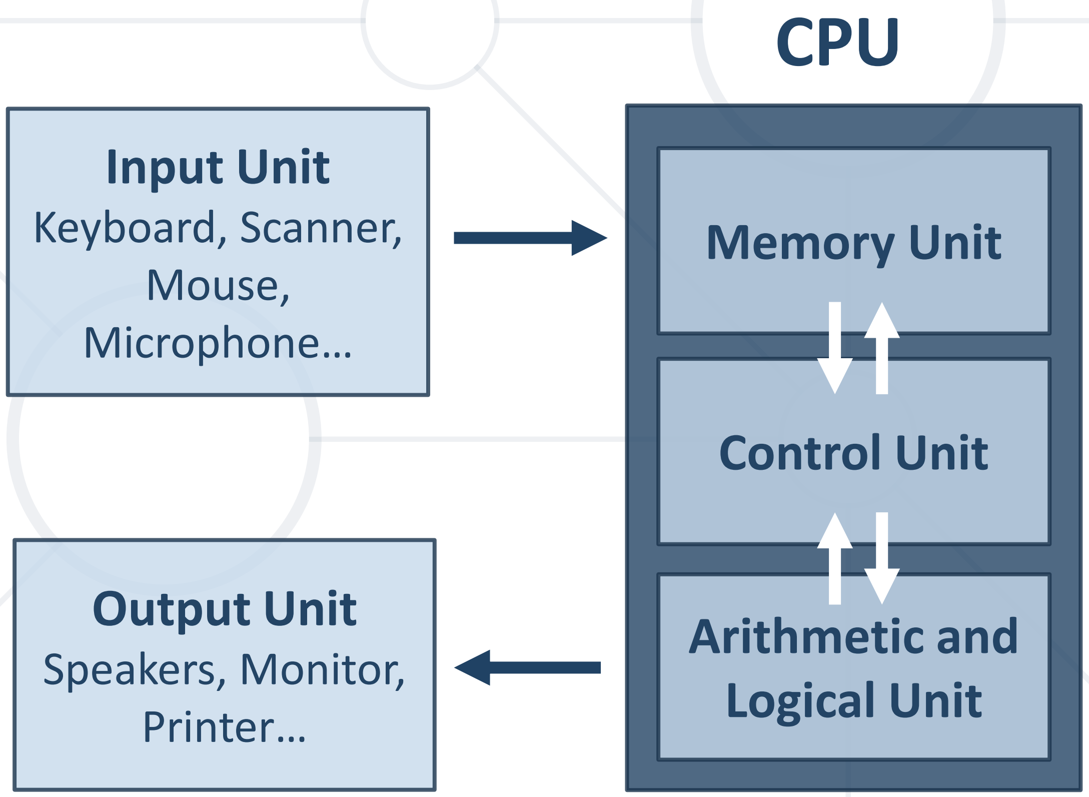
- **Early computing**: mechanical and electromechanical devices (e.g., Abacus, Babbage's Analytical Engine, ENIAC)
- **Advancements in technology**: transistors, integrated circuits, microprocessors (e.g., mainframe computers, minicomputers, personal computers)
- **Modern era**: pervasive computing, IoT, cloud computing, edge computing, rise of AI and machine learning

Computer Hardware

Motherboard, CPU, Memory, Storage, Peripherals

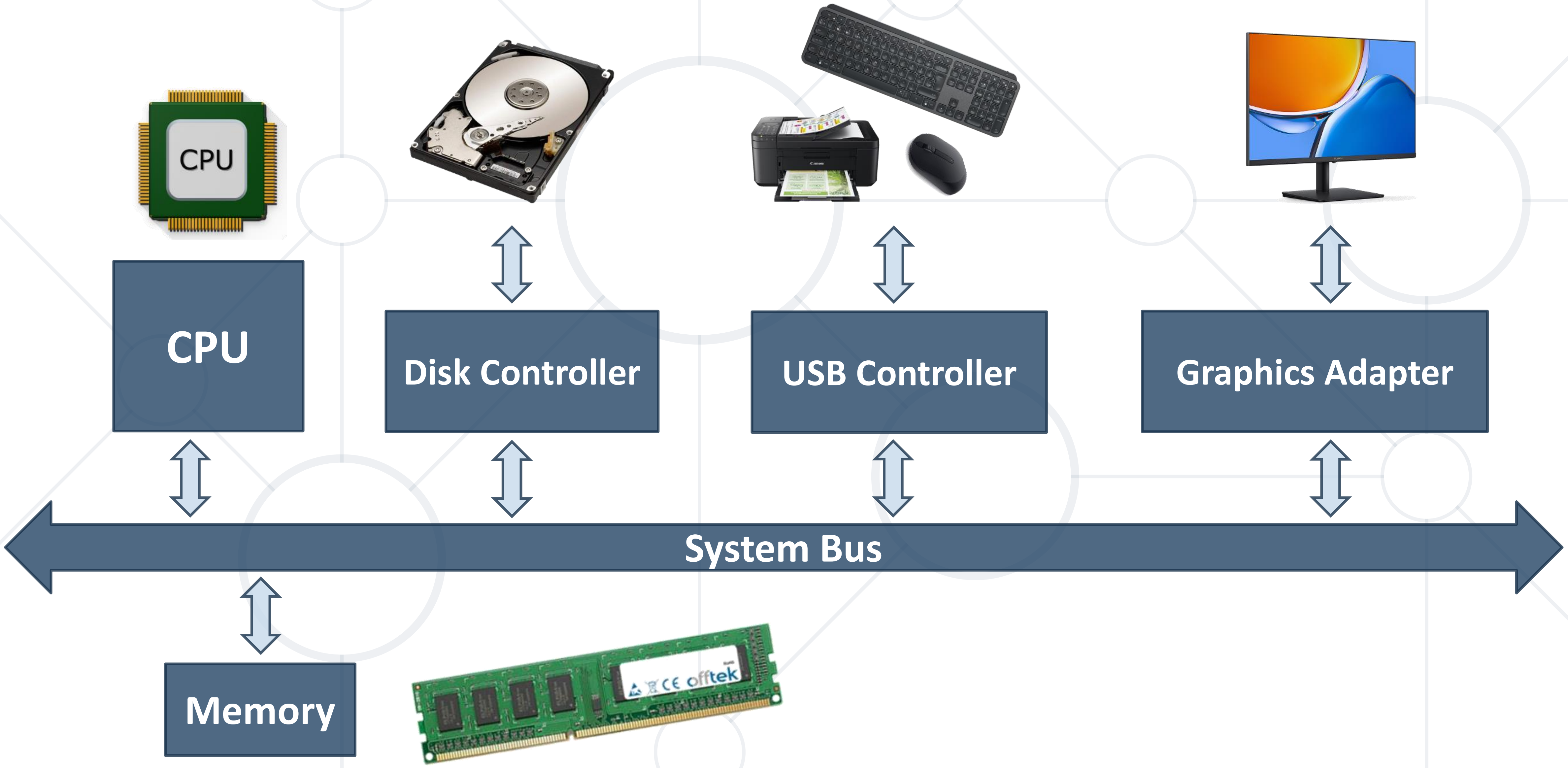
Computing Machines: Concepts

- **Hardware** refers to the **physical components** of a computer
- Central Processing Unit (**CPU**) – microprocessor
  - **Executes the code** (programs)
  - All **data processing operations**
- **Input devices**
  - Enter data
- **Output devices**
  - Get information



CPU
Input Unit
Keyboard, Scanner, Mouse, Microphone…
Memory Unit
Control Unit
Arithmetic and Logical Unit
Output Unit
Speakers, Monitor, Printer…

Computer System Hardware



Motherboard

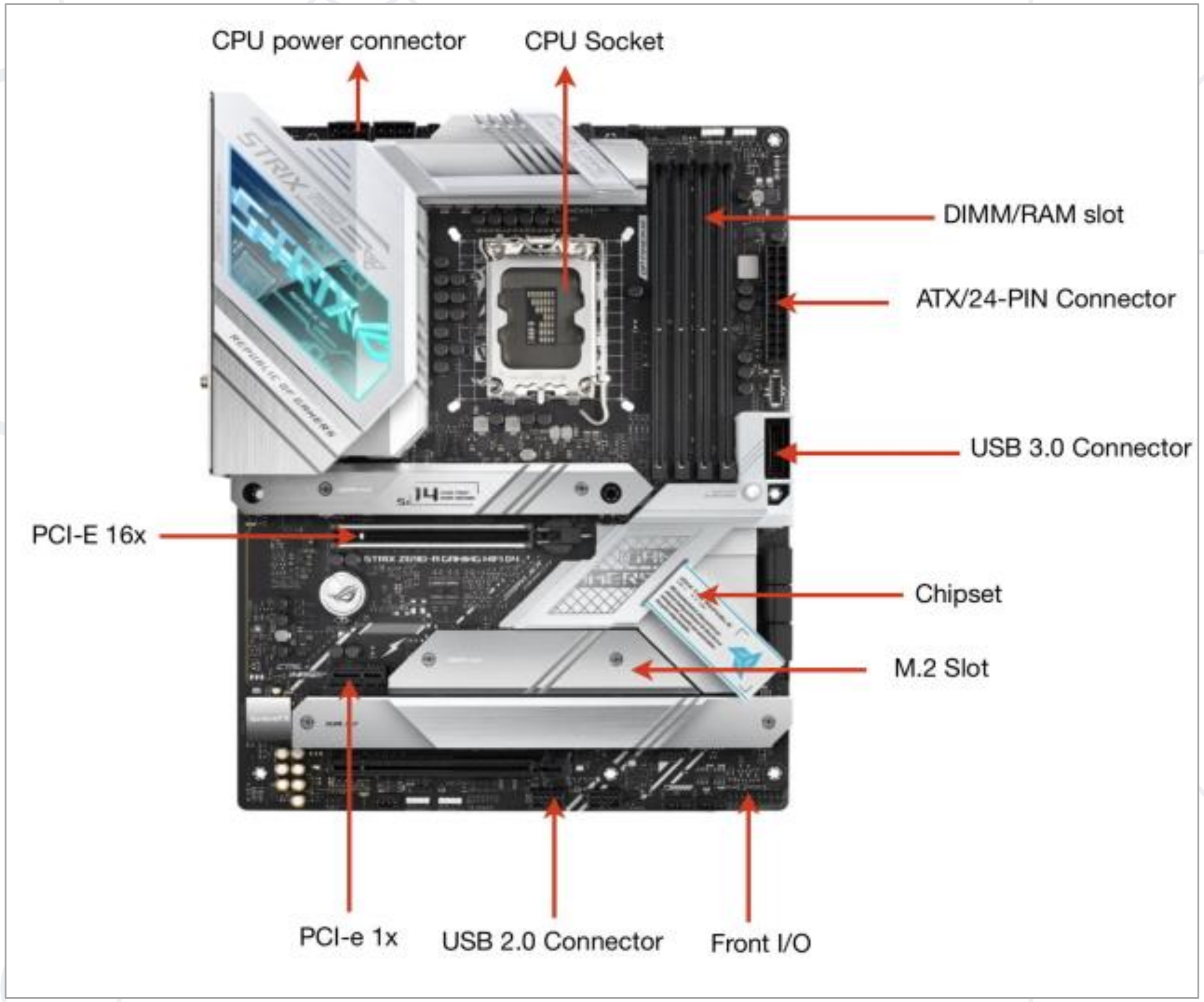
Backbone of a Computer System

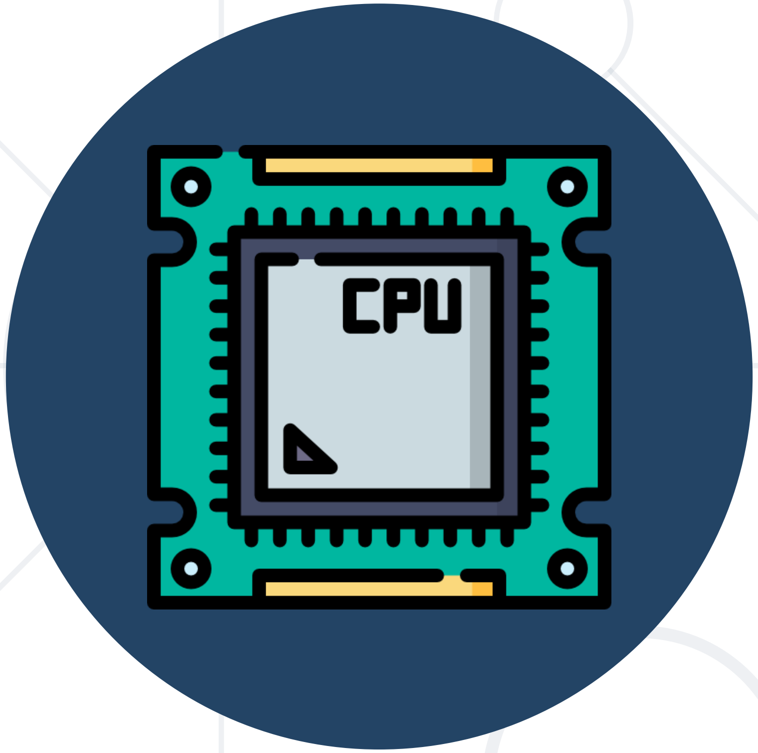
What is a Motherboard?

- **Motherboard** - central hub for **hardware connectivity**
  - **Communication** between all **hardware components**
- **Compatibility** considerations
  - Each motherboard is **designed to work with specific types** of **processors** and **memory**
- **Expansion slots** for enhanced functionality
  - **Video cards** for improved graphics performance
  - **Sound cards** for enhanced audio capabilities
  - **Network cards** for better internet connectivity

Motherboard Components

- CPU socket
- RAM slots
- Power connectors
- Chipset
- Expansion slots
- SATA connectors
- USB connectors
- Bluetooth module





CPU (Microprocessor)

Central Processing Unit

What is CPU?

- **CPU** – the brain of the computer
  - Executes **calculations**, **actions**, and **runs programs**
  - Provides **processing power** and **instruction control**
- Three **core components**
  - **Control Unit** (CU)
    - Manages instruction flow and coordinates hardware functions
  - **Arithmetic and Logical Unit** (ALU)
    - Performs arithmetic and logic operations
  - **Memory Unit** (MU)
    - Stores data, programs, and information

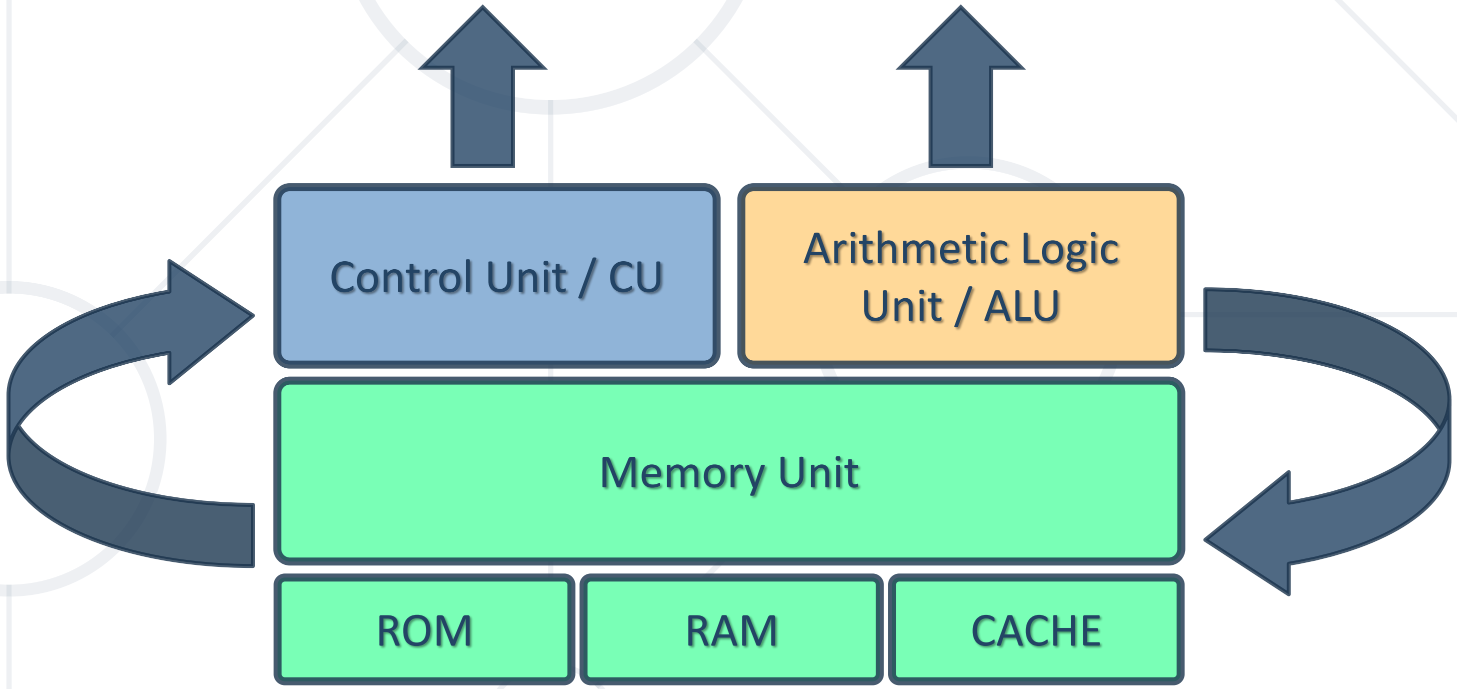
CPU Parts Workflow

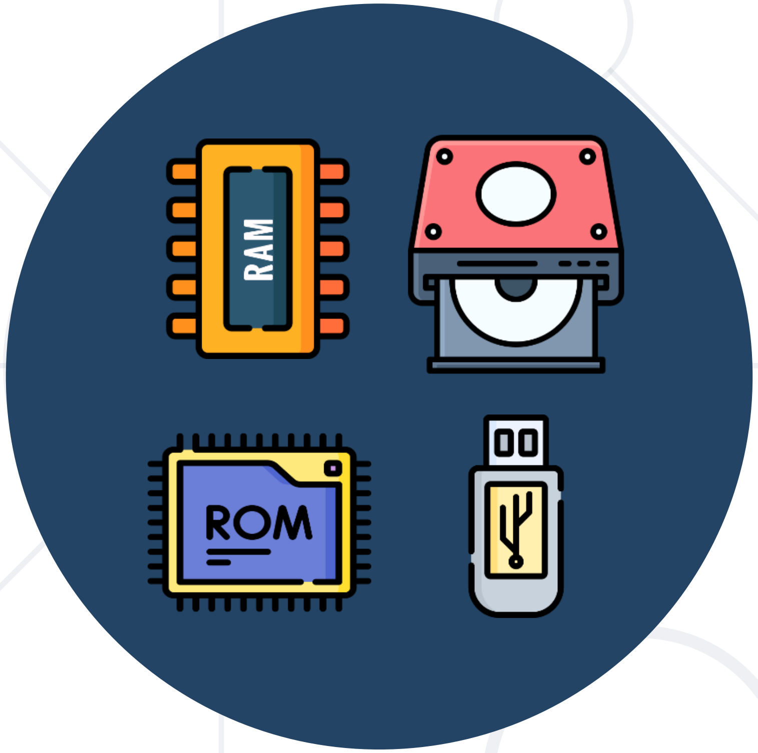
Step 2: Decode instructions into commands

Step 3: Execute commands

Step 1: Fetch instructions from the memory

Step 4: Store results in the memory





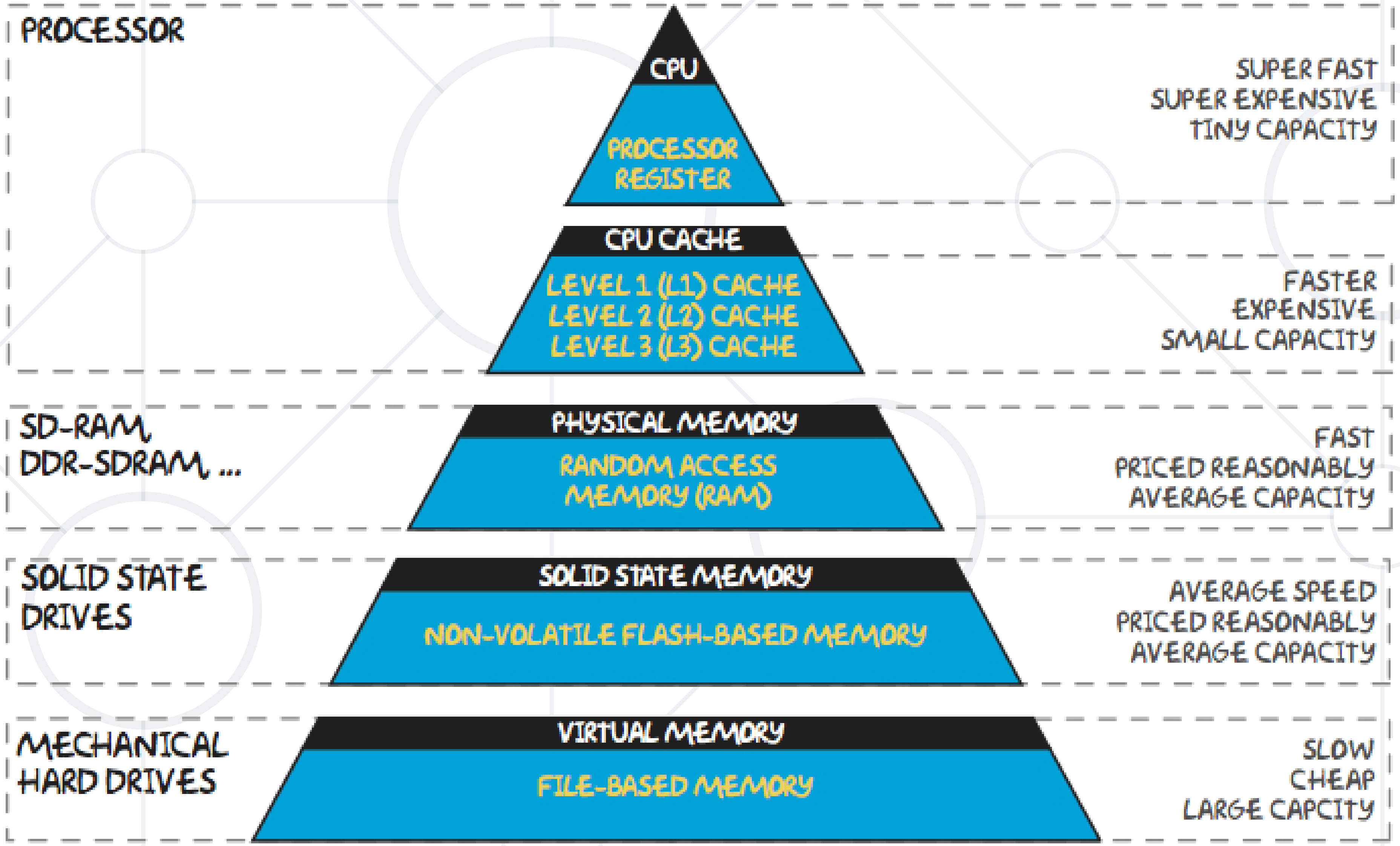
Memory and Storage

Storing Information in a Computer

Types of Memory

- **Primary memory**
  - **RAM** – read / write: stores data, required by the CPU during the **execution of a program**
  - **ROM** – read-only: stores **crucial data for the system** to operate, like the essential program for the computer boot
- **Secondary memory**
  - Not accessed directly by the processor
  - Examples: hard drive, SSD, flash, optical drive, USB drive
- **Cache memory**
  - Part of the CPU, very fast: temporarily stores **frequently used instructions and data** to speed-up access

Memory Hierarchy

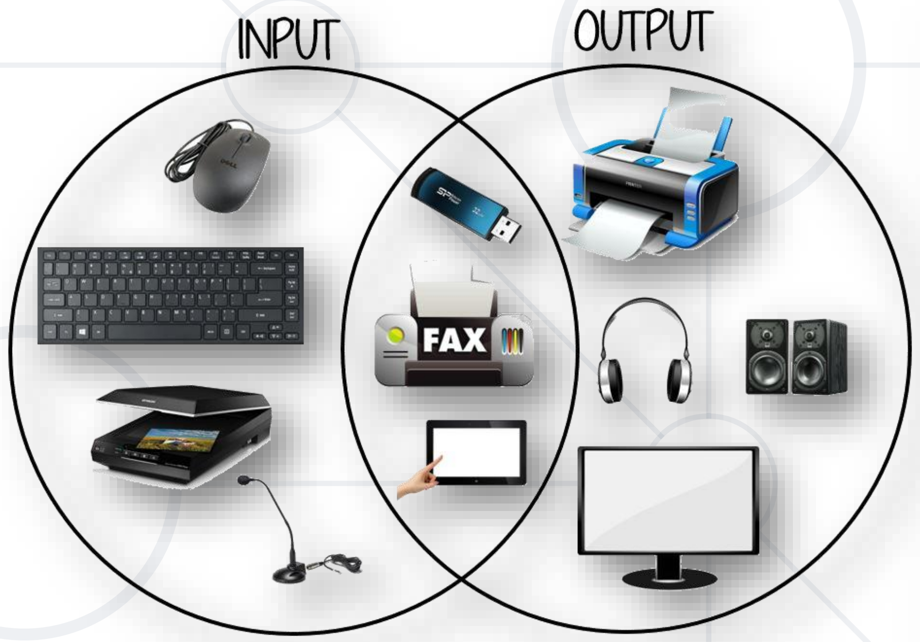


Peripheral Devices

Expanding Computer's Functionality

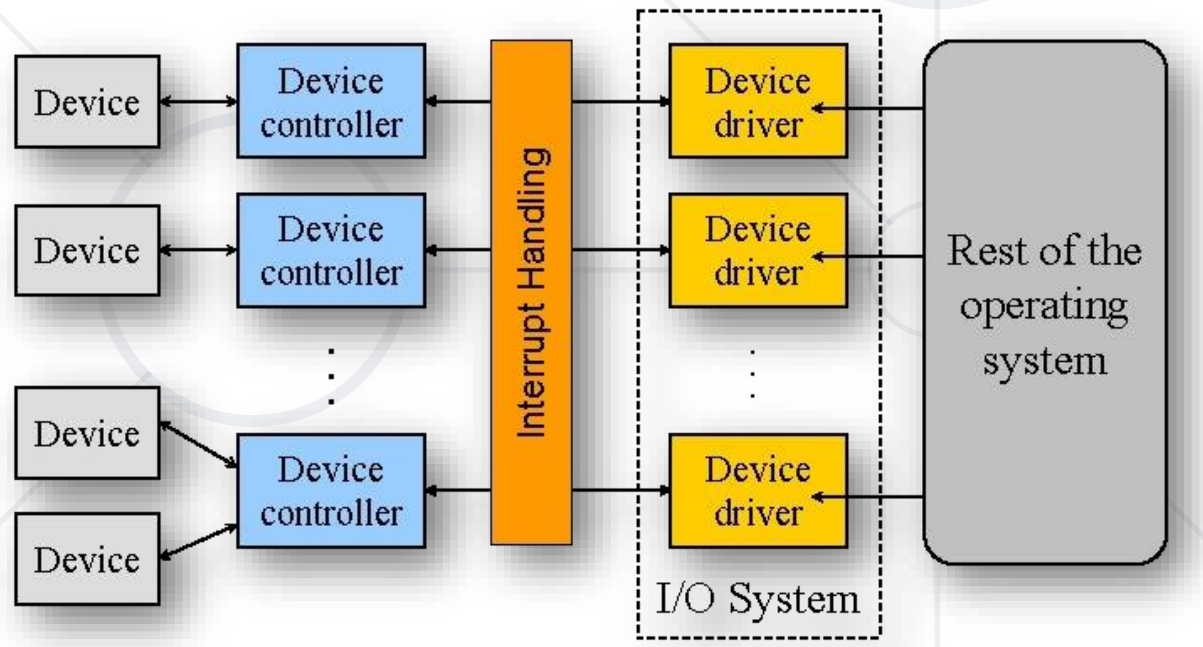
What is a Peripheral Device?

- **Any connected device** that expands computer's capabilities with **additional functionality**
- **Three main categories**:
  - **Input devices** → **read data**, e.g. keyboard, mouse, microphone
  - **Output devices** → **write data**, e. g. speakers, printer, monitor
  - **Input/output** devices → **mixed**, network card, hard drive, touchscreen monitor

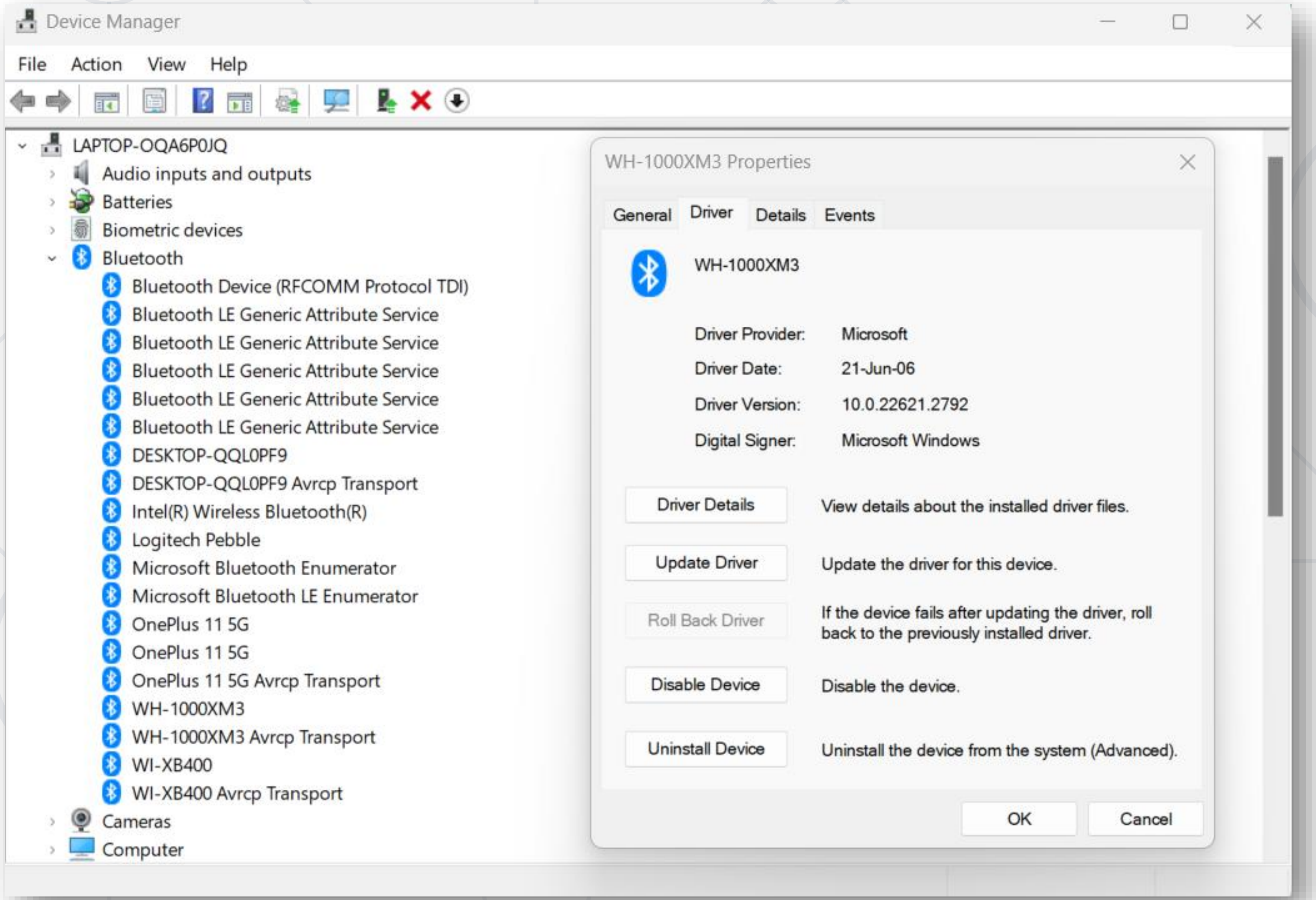


Peripheral Devices Control

- Device **controller**
  - A **physical device** for **connection** between a peripheral device and the computer, e. g. USB controller
- Device **driver**
  - **System software**, which enables the **communication and data transfer** between devices and the system



Device Manager in MS Windows



LAPTOP-OQA6P0JQ
- Audio inputs and outputs
- Batteries
- Biometric devices
- Bluetooth
  - Bluetooth Device (RFCOMM Protocol TDI)
  - Bluetooth LE Generic Attribute Service
  - Bluetooth LE Generic Attribute Service
  - Bluetooth LE Generic Attribute Service
  - Bluetooth LE Generic Attribute Service
  - Bluetooth LE Generic Attribute Service
  - DESKTOP-QQL0PF9
  - DESKTOP-QQL0PF9 Avrcp Transport
  - Intel(R) Wireless Bluetooth(R)
  - Logitech Pebble
  - Microsoft Bluetooth Enumerator
  - Microsoft Bluetooth LE Enumerator
  - OnePlus 11 5G
  - OnePlus 11 5G
  - OnePlus 11 5G Avrcp Transport
  - WH-1000XM3
  - WH-1000XM3 Avrcp Transport
  - WI-XB400
  - WI-XB400 Avrcp Transport
- Cameras
- Computer

WH-1000XM3 Properties

General | Driver | Details | Events

WH-1000XM3

Driver Provider:	Microsoft
Driver Date:	21-Jun-06
Driver Version:	10.0.22621.2792
Digital Signer:	Microsoft Windows

Driver Details	View details about the installed driver files.
Update Driver	Update the driver for this device.
Roll Back Driver	If the device fails after updating the driver, roll back to the previously installed driver.
Disable Device	Disable the device.
Uninstall Device	Uninstall the device from the system (Advanced).

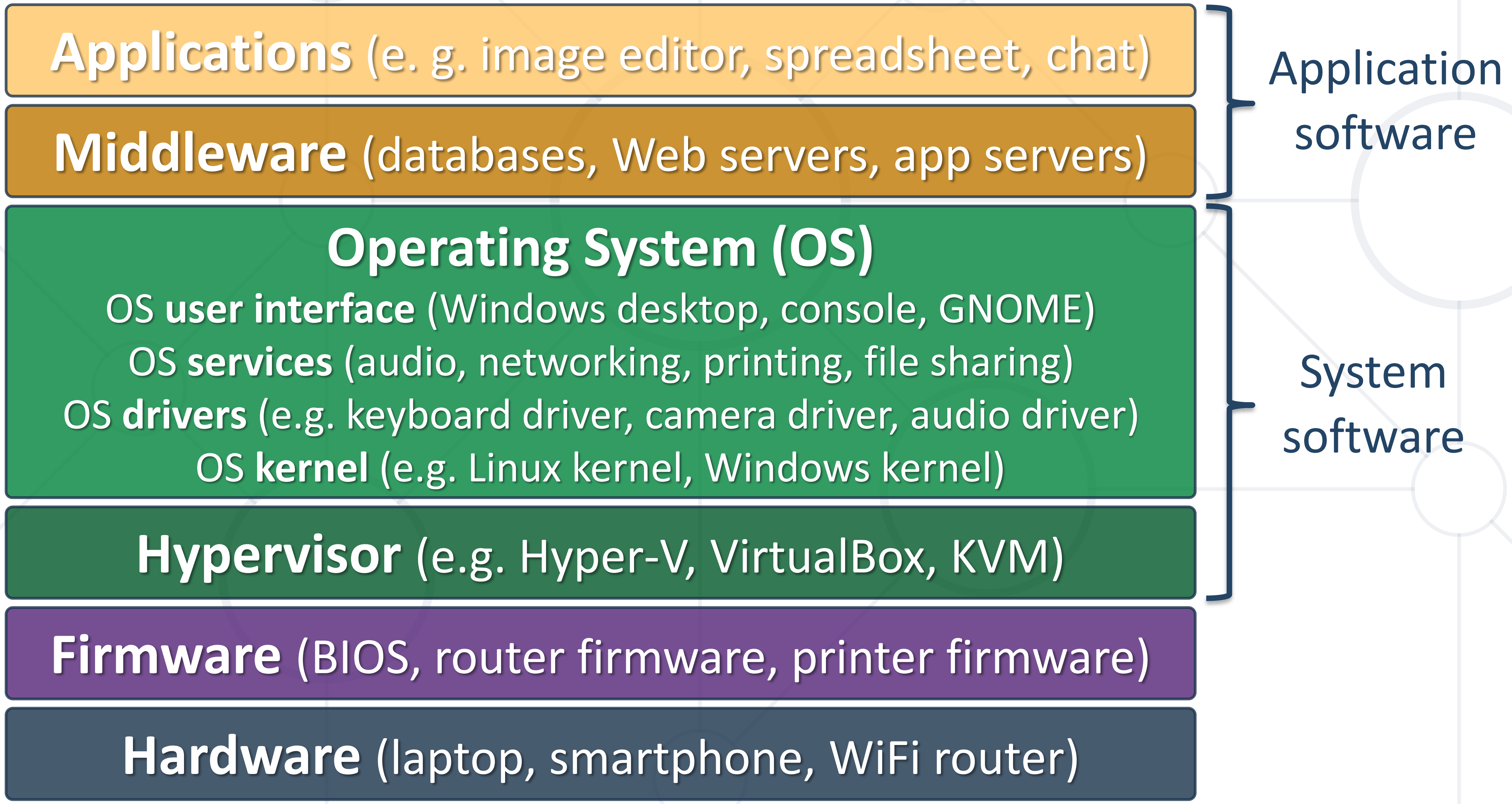
Computer Software

Firmware, System Software, Applications

Overview of Computer Software

- Computer software
  - Computer programs, instructions, and data that enable a computer system to **perform specific tasks**
- Types of software:
  - **Application software**: help the business to run, e.g. email software, spreadsheets, word processing, CRM systems, etc.
  - **System software**: interacts with and **manages the hardware**
- Standalone apps vs. software systems (client + server)

Software Stack


Applications (e. g. image editor, spreadsheet, chat)

Middleware (databases, Web servers, app servers)

Operating System (OS)
OS **user interface** (Windows desktop, console, GNOME)
OS **services** (audio, networking, printing, file sharing)
OS **drivers** (e.g. keyboard driver, camera driver, audio driver)
OS **kernel** (e.g. Linux kernel, Windows kernel)

Hypervisor (e.g. Hyper-V, VirtualBox, KVM)

Firmware (BIOS, router firmware, printer firmware)

Hardware (laptop, smartphone, WiFi router)

Application software

System software

Layers of Software

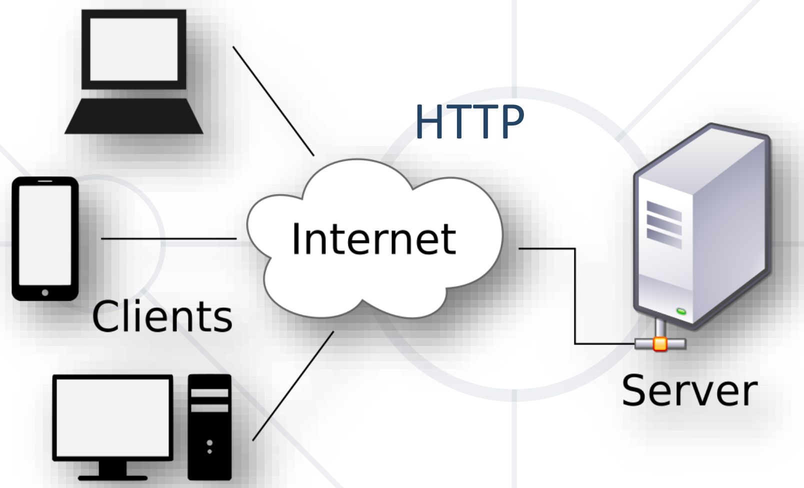
- **Firmware** and **embedded software**
  - **Low-level software** used to operate a hardware device
- **System software**
  - **Manages and controls hardware**, platform for applications
  - **Operating systems** (OS) – Windows, Linux, macOS, Android
  - **Hypervisors** – runs virtual machines (VMs) in the host OS
- **Application software**
  - Business applications, office apps, multimedia, communication
  - Several types: **Web apps**, **desktop apps**, **mobile apps**

Software Systems

- **Standalone apps**
  - Run **locally**, store their data locally, do not need Internet
  - Examples: Windows Calculator, Windows Explorer, Minesweeper
- **Software systems**
  - Consists of several **components** (e. g. client + server)
    - Example: mail server (remote) + mail client app (local)
  - **Cloud apps**: hold all user data in the cloud + local client
    - Example: Google Docs, Discord, Trello, Canva

Front-End and Back-End

- **Front-end** and **back-end** separate the modern apps into **client-side** (UI) and **server-side** (data) components
- **Front-end** == client-side components (Desktop / mobile app / Web browser)
  - Implement the **user interface** (UI)
- **Back-end** == server-side components (data and business logic APIs)
  - Implements **data storage and processing**



- **HTTP** connects front-end with back-end

Firmware

Bridge between Hardware and Software

What is Firmware?

- **Firmware** - **permanent**, low-level software, **embedded** in a device's read-only memory (ROM)
  - Controls device's basic functions and provides a **stable foundation** for higher-level software
  - Example: WiFi router's firmware, coffee machine firmware
- **Functions** of firmware
  - Hardware **initialization** during the boot process
  - **Management** of low-level hardware operations (e. g. device **initialization**, hardware **diagnostics**, and system **booting**)

Firmware: Devices and Use Cases

- Examples of **firmware applications**
  - BIOS / UEFI in laptops and desktop computers
  - Firmware in routers, printers, scanners
  - Embedded systems, such as IoT devices
- **Firmware updates**
  - Most devices allow firmware updates to improve functionality or fix issues
  - Can be critical for security and performance

System Software

Foundation for Application Software

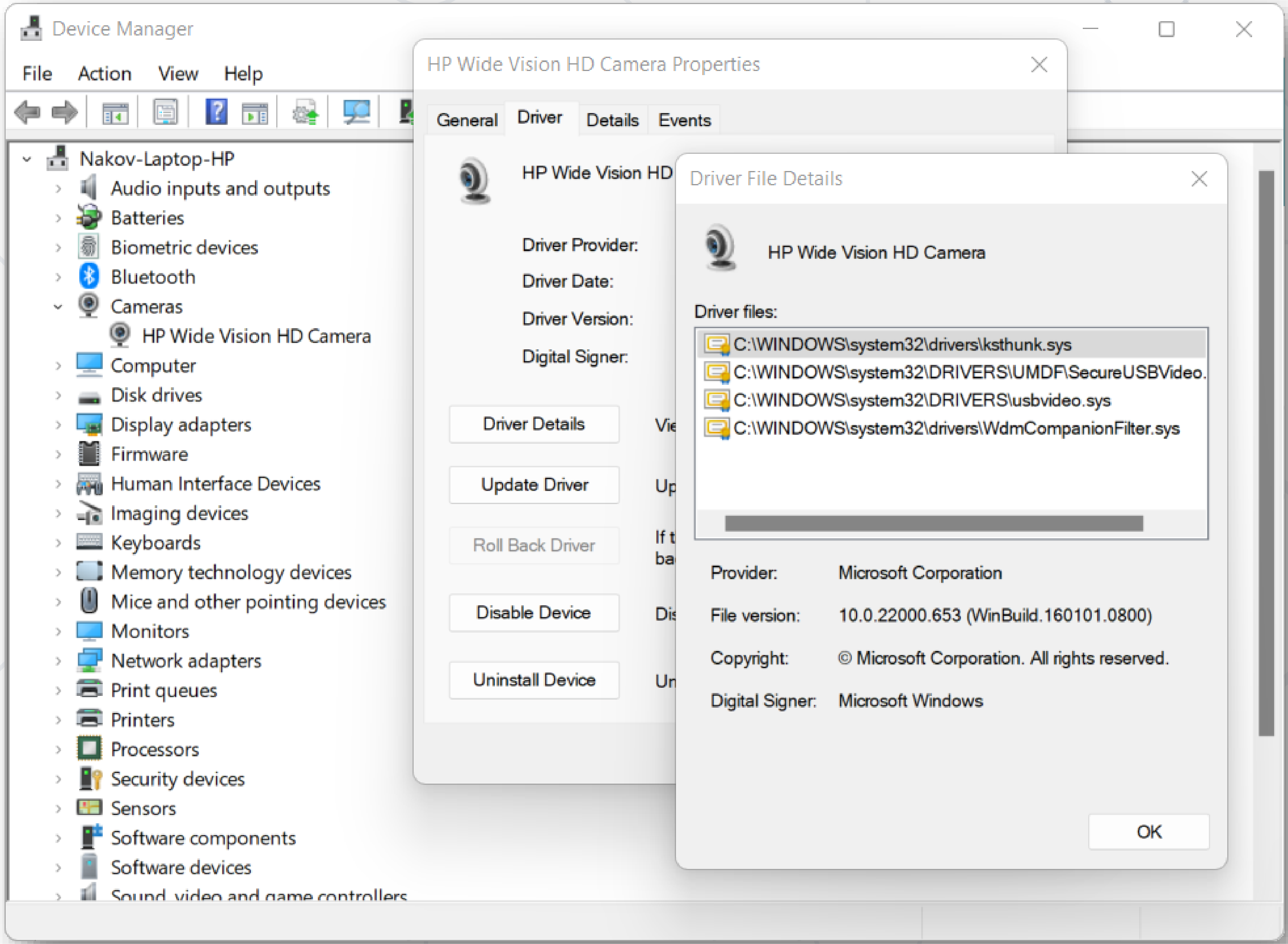
What is System Software?

- Software designed to **manage** and **control** computer **hardware**, providing a **platform** for **application software**
- Examples of system software:
  - **Hypervisors** – runs virtual machines (VMs) in the host OS
  - **Operating systems** (OS) – Windows, macOS, Linux, Android
  - **Device drivers** – software that enables communication between hardware and operating system), e. g. mouse driver
  - **System utilities** – tools for system maintenance and optimization, e. g. anti-virus, task manager, print spooler

Operating Systems

- Windows, macOS, Linux, Android, iOS
- Manage the **hardware** and **software** resources
- Manage **processes** (concurrently running apps)
  - Distribute the system resources between all processes
- Manage **file system** and **memory** (RAM)
- Manage **users**, **security** and **access control**
- System **updates** and **maintenance**

Device Drivers


Device Manager

File Action View Help

Nakov-Laptop-HP
- Audio inputs and outputs
- Batteries
- Biometric devices
- Bluetooth
- Cameras
  - HP Wide Vision HD Camera
- Computer
- Disk drives
- Display adapters
- Firmware
- Human Interface Devices
- Imaging devices
- Keyboards
- Memory technology devices
- Mice and other pointing devices
- Monitors
- Network adapters
- Print queues
- Printers
- Processors
- Security devices
- Sensors
- Software components
- Software devices
- Sound, video and game controllers

HP Wide Vision HD Camera Properties

General | Driver | Details | Events

HP Wide Vision HD

Driver Provider:

Driver Date:

Driver Version:

Digital Signer:

Driver Details | Update Driver | Roll Back Driver | Disable Device | Uninstall Device

Driver File Details

HP Wide Vision HD Camera

Driver files:

```
C:\WINDOWS\system32\drivers\ksthunk.sys
C:\WINDOWS\system32\DRIVERS\UMDF\SecureUSBVideo.
C:\WINDOWS\system32\DRIVERS\usbvideo.sys
C:\WINDOWS\system32\drivers\WdmCompanionFilter.sys
```

Provider:	Microsoft Corporation
File version:	10.0.22000.653 (WinBuild.160101.0800)
Copyright:	© Microsoft Corporation. All rights reserved.
Digital Signer:	Microsoft Windows

OK

- In Windows, the "**Device Manager**" lists all devices, drivers, etc.

System Utilities

- Tools that help maintain and optimize a computer system
  - **Antivirus** and **malware** protection (e.g. Winows Defender)
  - System **backup** and **recovery** (e. g. Macrium Reflect)
  - Disk **cleanup** and **defragmentation** (e. g. CCleaner)
  - Performance **monitoring** and **diagnostics** (Task Manager)
  - Software **updates** and **patches** (e. g. Windows Update)
  - System hardware **information** (e. g. CPU-Z)
  - System **logs viewer** (e. g. Windows Events Viewer)

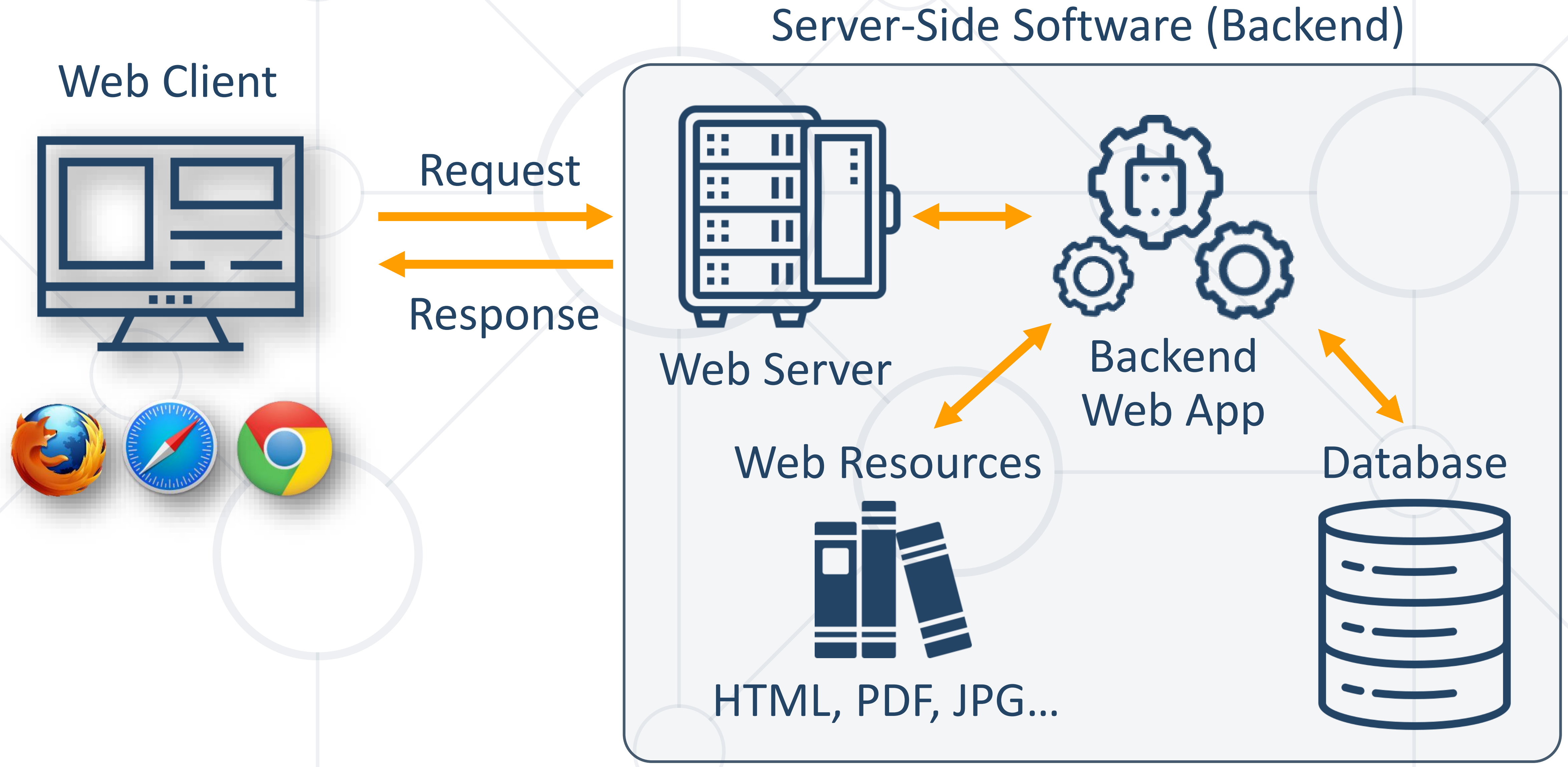
Server-Side Software (Backend)

Facilitating Backend Operations and Web Services

Server-Side Software Examples

- **Server-side software** (backend software) runs on a remote server, processes requests and delivers data to client devices
- **Common types** of server-side software
  - **Web** servers (e. g. Apache, Nginx, IIS)
  - **Database** servers (e. g. MySQL, PostgreSQL, MongoDB)
  - **Application** servers / runtimes (e. g. Tomcat, Node.js, .NET Core)
  - **Mail** servers (e. g. Microsoft Exchange Server, Postfix)
  - **File** servers (e. g. Windows File Server, Samba)
  - **Authentication servers** (e. g. FreeIPA, Active Directory)

The Client-Server Model in Web Apps



Server-Side vs. GUI

- **Server-side** software (backend software):
  - **Executes** on a **remote server**, rather than on the user's device
  - Handles **data processing, storage, and retrieval**
  - Powers Web applications, backend APIs, cloud services, etc.
  - Requires **efficient resource management** for optimal performance
- Graphical User Interface (**GUI**) / **front-end** apps:
  - **Executes** on the **user's device** (desktop, mobile, or Web)
  - Providing seamless and visually **appealing user experience**
  - Can be **Web** apps, **desktop** apps, or **mobile** apps

Application Software

Apps for the End Users

What is Application Software?

- **Application software** is designed for users to perform **specific business tasks**, catered to their **individual needs**
- Examples of application software
  - **Productivity** tools (Microsoft Office, Google Workspace)
  - **Multimedia** software (Adobe Photoshop, VLC Media Player)
  - **Communication** apps (Zoom, WhatsApp, MS Teams)
  - **Web browsers** (Google Chrome, Mozilla Firefox, Safari)
  - **Games** (Fortnite, League of Legends)

Web Apps

Applications, Accessed from the Web Browser

Web Applications

- What are **Web apps**?
  - Accessed through a **Web browser** with an **active Internet** connection
  - **Platform-independent**
    - Accessible on any device with a Web browser
    - Desktop / mobile Web browsers
  - **Automatic updates** (always up-to-date)
    - No need for manual installation or updating

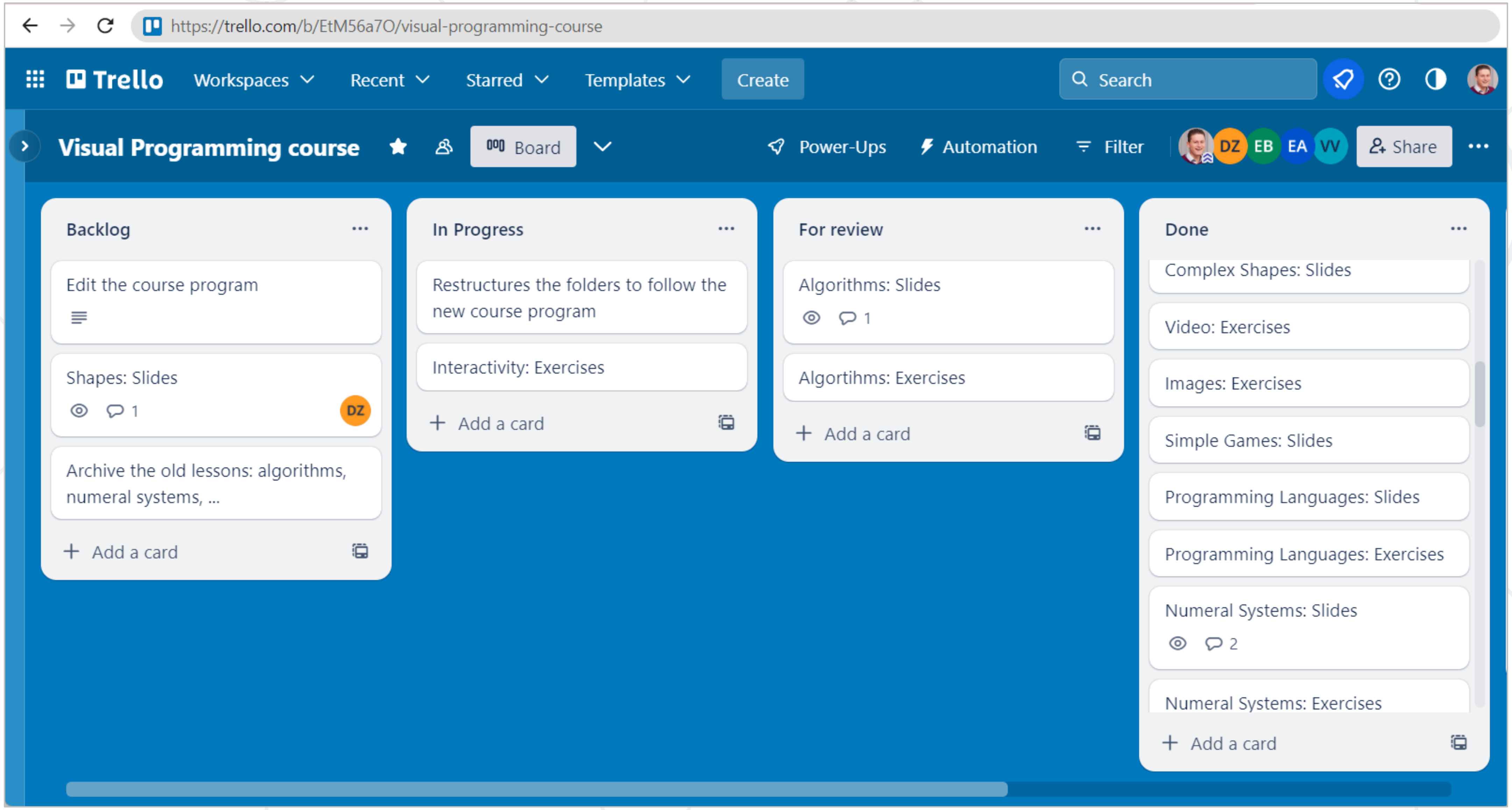
Web Applications Benefits

- Benefits of **Web apps**
  - **Scalability**: easily accommodate a growing user base
  - **Centralized data storage**: simplifies data management and backup
  - **Lower device requirements**: minimal hardware needed (processing is done on the server-side)
  - **Easier collaboration**: real-time collaboration
  - **Cross-platform compatibility**: works across various operating systems and devices

Testing Challenges for Web Apps

- **Compatibility** - if the app works consistently across **different Web browsers** and different **screen sizes** (responsive design)
- **Usability** - testing for **accessibility**, **intuitive use** on different devices, and ease of **navigation**
- **Network conditions** - Web apps rely on an active internet connection→ testing under **different network conditions**
- **Security** - Web apps deal with sensitive data → testing for **vulnerabilities** such as XSS attacks and SQL injection
- **Performance** - performance can be affected by network speed / server load / browser capabilities → testing for **scalability** / **load capacity**

Trello Project Management Web App


https://trello.com/b/EtM56a7O/visual-programming-course

Backlog
- Edit the course program
- Shapes: Slides — 1 — DZ
- Archive the old lessons: algorithms, numeral systems, …
- + Add a card

In Progress
- Restructures the folders to follow the new course program
- Interactivity: Exercises
- + Add a card

For review
- Algorithms: Slides — 1
- Algortihms: Exercises
- + Add a card

Done
- Complex Shapes: Slides
- Video: Exercises
- Images: Exercises
- Simple Games: Slides
- Programming Languages: Slides
- Programming Languages: Exercises
- Numeral Systems: Slides — 2
- Numeral Systems: Exercises
- + Add a card

Desktop Apps

Applications Running Locally on Your Laptop

Desktop Applications

- What are **desktop apps**?
  - **Installed** and **run locally** on a user's computer
    - Store their data locally or remotely (depends)
  - **Offline access**
    - Can be used without an Internet connection
  - **More features**
    - Often more feature-rich than Web apps
    - Better integrated with the host OS

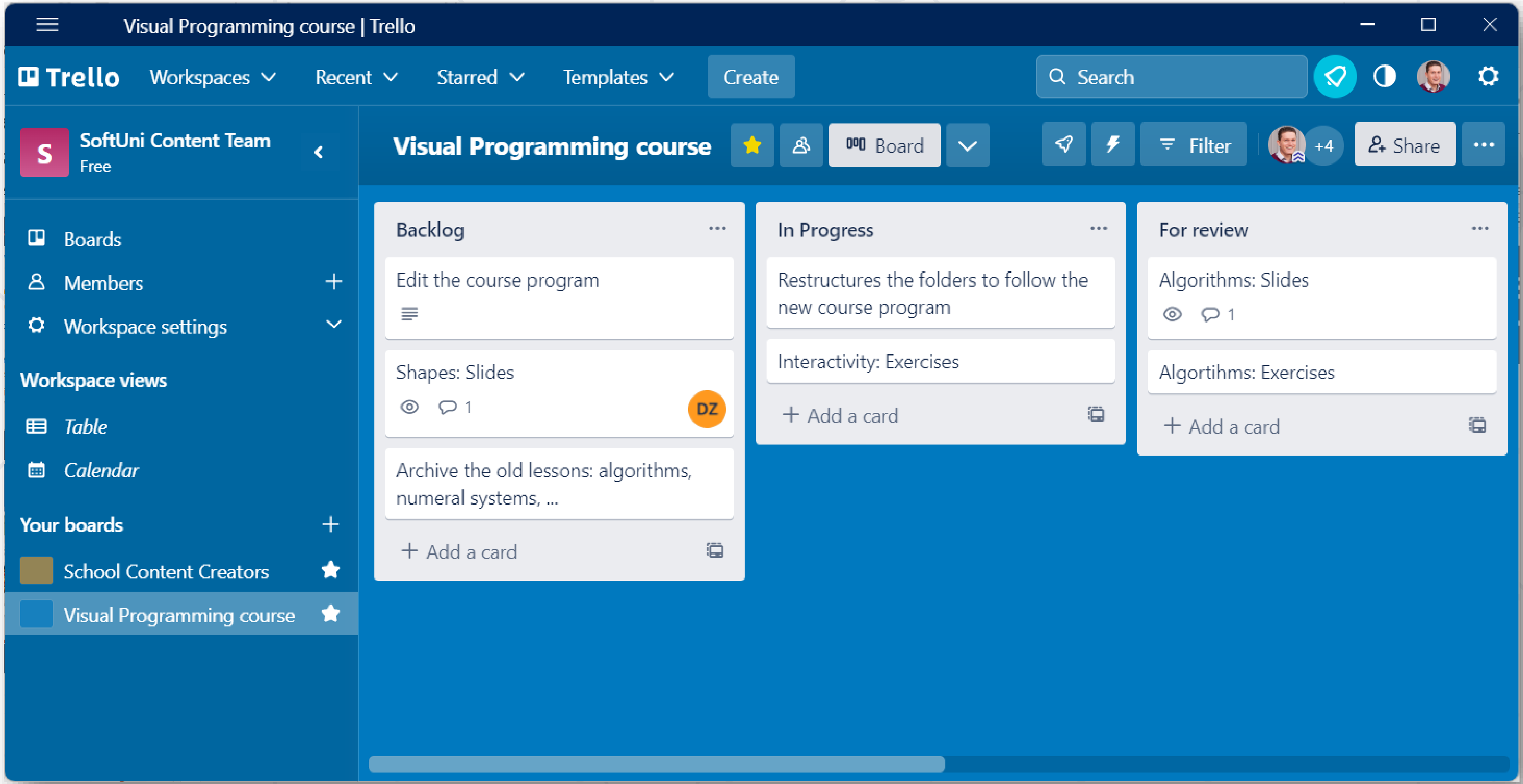
Desktop Applications Benefits

- Benefits of **desktop apps**
  - **Performance**: faster processing and response time, as tasks are executed locally
  - **Customization**: easily tailored to individual user preferences and needs
  - **Integration**: compatible with other locally installed software and hardware
  - **Cost-effective**: one-time purchase or licensing fees, instead of recurring subscription costs (depends)

Testing Challenges for Desktop Apps

- **Installation / uninstallation** including any **dependencies** or **prerequisites**
- **Performance testing on different hardware configurations** – **processors**, **memory**, and **graphic cards**
- **Compatibility testing** for **different operating systems** and their **different versions**
- **User interface testing** - desktop apps often have complex **UI** that need to be thoroughly tested
- **Integration testing** with **other desktop applications**

Trello Project Management Desktop App


Visual Programming course | Trello

SoftUni Content Team
Free

School Content Creators
Visual Programming course

Visual Programming course

Backlog
- Edit the course program
- Shapes: Slides (1) DZ
- Archive the old lessons: algorithms, numeral systems, ...
- + Add a card

In Progress
- Restructures the folders to follow the new course program
- Interactivity: Exercises
- + Add a card

For review
- Algorithms: Slides (1)
- Algortihms: Exercises
- + Add a card

Mobile Apps

Applications Running Locally on Mobile Device

Mobile Applications

- What are **mobile apps**?
  - Designed specifically for **smartphones** and **tablets**
  - Accessible through **dedicated app stores** (e.g., Google Play, Apple App Store)
  - Optimized for **touchscreen interfaces** and mobile device features (adaptable UI design for different screen sizes)
  - Can work **offline**, **online** or mixed

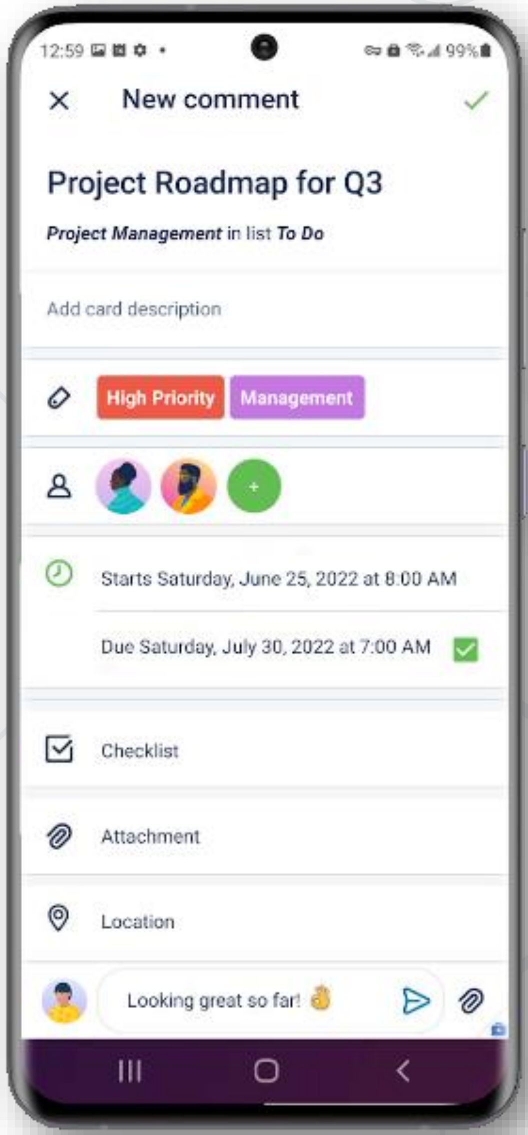
Mobile Applications Benefits

- Benefits of **mobile apps**
  - **Portability**: access apps and data on-the-go, anytime, anywhere
  - **Push notifications**: real-time updates and alerts for improved user engagement
  - **Device-specific features**: leverage device capabilities like GPS, camera, and sensors
  - **Offline functionality**: some apps can operate without an Internet connection
  - **Streamlined user experience**: tailored for smaller screens and touch-based interactions

Testing Challenges for Mobile Apps

- **Compatibility** across different devices and OS versions is crucial for mobile apps (**many different devices** and **versions in use**)
- **User interface testing** – **design and layout** has significant impact on the user's experience on a **smaller screen**
- **Performance testing** – performance may be affected by **limited processing power** and **memory** on the user's device
- **Battery life testing** – to ensure that the app does not significantly **drain** the user's device **battery**

Trello Project Management Mobile App



12:59 99%

New comment

Project Roadmap for Q3

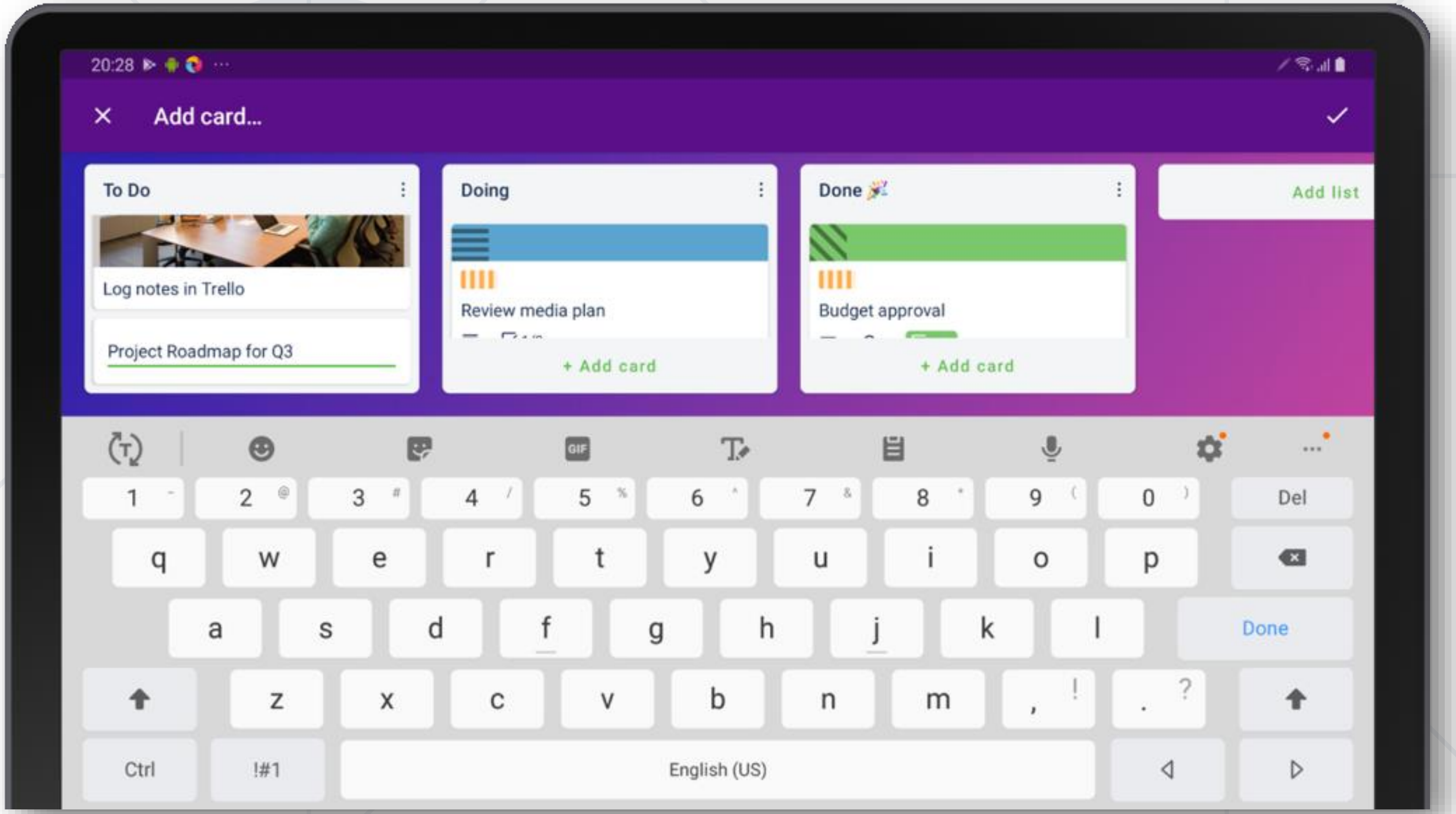
Project Management in list *To Do*

Add card description

Starts Saturday, June 25, 2022 at 8:00 AM

Due Saturday, July 30, 2022 at 7:00 AM ✓

Looking great so far! 👌



20:28

Add card...

To Do	Doing	Done 🎉	Add list
Log notes in Trello	Review media plan	Budget approval	
Project Roadmap for Q3	+ Add card	+ Add card	

Summary

- Hardware is the physical part
  - Main computer parts: motherboard (ties together all components), CPU (code execution), input / output devices
- Software – programs, running in the computer
  - Firmware and system software (OS, hypervisors)
  - Server-side software (back-end) vs. GUI / front-end apps
  - Application software (end-user apps): Web, Desktop, Mobile apps
  - Software systems (client + server) and cloud apps

Questions?

Software University

SoftUni

Software University

SoftUni Creative

SoftUni Digital

SoftUni Foundation

SoftUni Kids

Finance Academy

SoftUni Diamond Partners

Software University

SmartIT

Coca-Cola HBC Bulgaria

SOFTWARE GROUP

DXC TECHNOLOGY

createX

Postbank

Решения за твоето утре

SUPER HOSTING .BG

DRAFT KINGS

VIVACOM

INDEAVR

Serving the high achievers

AMBITIONED

PHAR VISION

License

Trainings @ Software University (SoftUni)

- Software University – High-Quality Education, Profession and Job for Software Developers
  - softuni.bg, about.softuni.bg
- Software University Foundation
  - softuni.foundation
- Software University @ Facebook
  - facebook.com/SoftwareUniversity
- Software University Forums
  - forum.softuni.bg