Hardware and software design constraints for mobile applications

Hardware and software design constraints for mobile applications include memory, battery life, screen size, security, network bandwidth, and user experience.

Hardware constraints

- **Memory**: Limited RAM and storage capacity can impact performance
- **Battery life**: Battery life is a key constraint for mobile applications

Software constraints

- **User experience**: A well-designed UI can improve the user experience
- **Security**: Mobile devices are attractive targets for cyber threats
- **Network bandwidth**: Network bandwidth can impact performance

Design strategies

- **Compatibility and interoperability**: Identifying and addressing compatibility issues early in design can reduce costs and delays
- **Resource constraints**: Developers can optimize code, upgrade hardware, or implement load balancing to improve performance
- **User experience**: A well-designed UI can improve the user experience
- **Security**: Mobile devices are attractive targets for cyber threats

Other considerations

- **Operating systems**: Mobile operating systems allow smartphones and tablets to run apps and perform other tasks
- **App architecture**: The structural components and interfaces that comprise a mobile app system

Should it be a native client, a thin web client, or rich internet application (RIA)?

If your application requires local processing and needs to occasionally work in a connected scenario, consider designing it as a native client, although it will be more complex to deploy and maintain. If your application is highly dependent on server processing and will always be fully connected, consider designing it as a thin client. If your application requires a rich user interface (UI), only limited access to local resources, and must be portable to other platforms, design as a RIA client.

Design for occasionally connected, limited-bandwidth scenarios

Mobile devices are standalone devices, which means you will need to account for connection issues. It is common for a mobile application to fall into a use case where network connection is intermittent or not available. Hence it is important to design for application caching, state management, and data access mechanisms with intermittent network connectivity in mind. One very effective approach to handling this use case is to maintain content locally (offline mode) with server sync state; then as soon as a network connection is established, to push local content to the server. Another approach is batch communication for times of connectivity. Also choose hardware and software protocols based on speed, power consumption, and “chattiness”—not just on ease of programming.

Design a UI appropriate for mobile devices, taking into account platform constraints

Mobile devices require a simpler architecture, simpler UI, and other specific design decisions in order to work within the constraints imposed by the device hardware. Keep these constraints in mind and design specifically for the device instead of trying to reuse the architecture or UI design from a desktop or web application. The main constraints are memory, battery life, ability to adapt to different screen sizes and orientations, security, and network bandwidth.

Design a layered architecture appropriate for mobile devices that improves reuse and maintainability

Depending on the application type, multiple layers can be located on the device itself. Use the concept of layers to maximize separation of concerns and to improve reuse and maintainability for your mobile application. However, aim to achieve the smallest footprint on the device by simplifying your design compared to a desktop or web application.

Consider device resource constraints such as battery life and memory size

Every design decision should take into account the memory usage and battery life of mobile devices. With processors such as Snapdragon 800 2.3GHz quad-core and RAM sizes of 2-3 GB already on the market, processor speed and available memory are not major concerns in mobile application design today. Battery life is usually the most limiting factor in mobile devices. Backlighting, reading and writing to memory, wireless connections, network traffic, and specialized hardware such as GPS have an impact on overall power usage. Although amount of memory available is not the key constraint, exploitation of memory may lead operating system to ask your application to shut down or sacrifice cached data, slowing program execution. Optimize your application to minimize its power and memory footprint while considering performance during this process.