

Mastering the Android System Design Interview

Useful Links Collection

Networking

HTTP Evolution:

- HTTP/1.0: <https://tools.ietf.org/html/rfc1945>
- HTTP/2.0: <https://ably.com/topic/http2>
- HOL: <https://engineering.cred.club/head-of-line-hol-blocking-in-http-1-and-http-2-50b24e9e3372>
- HTTP/3.0: <https://ably.com/topic/http3>
- Persistent & pipelined connections: https://www.youtube.com/watch?v=JJmVEI_Fkc
- Virtual Hosting: https://en.wikipedia.org/wiki/Virtual_hosting
- Content negotiation: https://developer.mozilla.org/en-US/docs/Web/HTTP/Content_negotiation
- Transfer encoding: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Transfer-Encoding>
- Content encoding: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Encoding>
- Caching: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Caching#validation>
- Authentication:
 - <https://developer.mozilla.org/en-US/docs/Web/HTTP/Authentication>
 - <https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/WWW-Authenticate>
- CORS: <https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>

Client-Server Communication:

- Long Polling
 - <https://www.youtube.com/watch?v=k56H0DHqu5Y>
 - <https://www.educative.io/edpresso/what-is-http-long-polling>
- WebSockets & Server-Sent Events
 - <https://medium.com/system-design-blog/long-polling-vs-websockets-vs-server-sent-events-c43ba96df7c1>
 - <https://ably.com/topic/websockets>
 - <https://datatracker.ietf.org/doc/html/rfc6455>
 - <https://ably.com/topic/server-sent-events>
 - <https://ably.com/blog/websockets-vs-sse>
 - <https://medium.com/@manoel.al.amaro/understand-kotlin-flow-coroutines-by-implementing-server-side-sse-9e190ff5f24f>
- HTTP Streaming: <https://datatracker.ietf.org/doc/html/rfc6202#section-3>
- REST
 - <https://restfulapi.net/>
 - https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm
 - <https://blog.restcase.com/http2-benefits-for-rest-apis/>
- gRPC
 - <https://grpc.io/docs/what-is-grpc/introduction/>
 - <https://www.imaginarycloud.com/blog/grpc-vs-rest/>
- GraphQL: <https://graphql.org/learn/best-practices/>
- FCM: <https://firebase.google.com/docs/cloud-messaging/fcm-architecture>
- Pagination
 - <https://developer.android.com/topic/libraries/architecture/paging/v3-overview>
 - <https://nordicapis.com/everything-you-need-to-know-about-api-pagination/>
 - <https://ignaciochiazzo.medium.com/paginating-requests-in-apis-d4883d4c1c4c>
 - <https://medium.com/swlh/how-to-implement-cursor-pagination-like-a-pro-513140b65f32>
 - <https://uxdesign.cc/why-facebook-says-cursor-pagination-is-the-greatest-d6b98d86b6c0>
- Background Services: <https://medium.com/@Codeible/understanding-and-using-services-in-android-background-foreground-services-8130f6bbf2a5>
- Web Transport: <https://ably.com/blog/can-webtransport-replace-websockets>
- XMPP: <https://www.mirrorfly.com/blog/xmpp-vs-websockets-instant-messaging-protocol-comparison/>

Storage

- File Storage
 - <https://developer.android.com/guide/topics/data>
 - <https://developer.android.com/training/data-storage>
- Preferences
 - <https://developer.android.com/training/data-storage/shared-preferences>
 - <https://proandroiddev.com/is-jetpack-datastore-a-replacement-for-sharedpreferences-efe92d02fcb3>
 - <https://android-developers.googleblog.com/2020/09/prefer-storing-data-with-jetpack.html>
 - <https://developer.android.com/topic/libraries/architecture/datastore>
- Databases
 - <https://developer.android.com/training/data-storage/room>
 - <https://github.com/donnemartin/system-design-primer#nosql>
 - <https://github.com/donnemartin/system-design-primer#sql-or-nosql>
 - <https://www.tutorialspoint.com/what-is-the-difference-between-blob-and-clob-datatypes>
 - <https://dba.stackexchange.com/questions/2445/should-binary-files-be-stored-in-the-database>
- Android Secure Keystore
 - <https://developer.android.com/topic/security/best-practices#safe-data>
 - <https://developer.android.com/training/articles/keystore>
 - <https://medium.com/@josiassena/using-the-android-keystore-system-to-store-sensitive-information-3a56175a454b>

Building for Billions

- Connectivity
 - <https://developer.android.com/docs/quality-guidelines/build-for-billions/connectivity>
 - <https://developer.android.com/training/connectivity/network-access-optimization#radio-state>
 - <https://developer.android.com/training/connectivity/troubleshoot-network-issues#traffic-types>
 - <https://developer.android.com/training/connectivity/minimize-effect-regular-updates#app-initiated>
- Offline Mode: <https://tech.trello.com/sync-architecture/>
- Data cost: <https://developer.android.com/docs/quality-guidelines/build-for-billions/data-cost>
- Device capability: <https://developer.android.com/docs/quality-guidelines/build-for-billions/device-capacity>
- Battery consumption
 - <https://developer.android.com/docs/quality-guidelines/build-for-billions/battery-consumption>
 - <https://instagram-engineering.com/improving-performance-with-background-data-prefetching-b191acb39898>
 - <https://github.com/weeeBox/mobile-system-design#resumable-uploads>
 - <https://developer.android.com/training/monitoring-device-state/battery-monitoring>
- Doze Mode & App Standby:
 - <https://developer.android.com/training/monitoring-device-state/doze-standby>
 - <https://developer.android.com/topic/performance/power/power-details>
- UI
 - <https://developer.android.com/docs/quality-guidelines/build-for-billions/ui>
 - <https://developer.android.com/training/improving-layouts>
 - <https://developer.android.com/topic/performance/rendering/overdraw.html>
- App Startup
 - <https://developer.android.com/topic/performance/vitals/launch-time>
- Memory
 - <https://developer.android.com/topic/performance/memory-overview>
 - <https://developer.android.com/topic/performance/memory-management>
 - <https://developer.android.com/topic/performance/memory>
 - <https://www.youtube.com/watch?v=w7K0jio8afM>
- Background work: <https://developer.android.com/guide/background>

General Guidelines

- <https://proandroiddev.com/a-simple-framework-for-mobile-system-design-interviews-89f6f4134b84>
- <https://medium.com/geekculture/system-design-interview-for-mobile-engineers-ce712d6ac2c1>
- <https://dev.to/ecaselles/cracking-the-mobile-system-design-interview-ios-android-4kfi>