UNIT 3 BASICS OF MOBILE APPLICATION DESIGN

Structure

3.0 INTRODUCTION

Designing an impactful mobile app involves various aspects. The designer should consider various elements such as user interface, design best practices, optimal integration methodologies, patterns, security, etc. Successful roll out of a mobile app also involves effective testing and robust deployment practices.

In this chapter, we will look at key design best practices, deep dive of user interface design, security aspects and testing elements of a mobile app.

We also provide a checklist for mobile apps and touch base upon power usage and synchronization aspects of a mobile app.

3.1 OBJECTIVES

After going through this unit, you should be able to:

- understand key design considerations and best practices of a mobile app,
- know the anti-patterns of mobile app design,
- deeper understanding of user interface design of mobile app,
- know the deployment and power usage scenarios,
- Know the security standards, and
- Know various kinds of testing

3.2 DESIGN CONSIDERATIONS AND BEST PRACTICES

The key design considerations for mobile applications are as follows:

- Firstly, decide on the type of the mobile app – web, native or hybrid.
- Design the mobile application considering various form factors, screen sizes, orientations and resolutions. Test for all supported devices.
- Design the code to use the device memory, battery and storage optimally.
- Provide mobile friendly controls, navigation and touch enabled actions.
- Provide support for multiple languages and font sizes.
- Design fast, responsive and interactive page layouts.

The following are the key best practices for designing mobile applications:

- Design applications to handle lost network using offline features.
- Test the mobile applications for slow performance scenarios and handle the transactions gracefully.
- Use the least restrictive security model and provide permissions to the mobile apps only when required and when it is permitted by the user.
- Utilize notification features, events, messaging and progress bars wherever necessary.
- Mandatorily encrypt data during transmission.
- Mobile app should leverage offline caching and local caching for rich and interactive apps.
- Given below are the user experience related best practices:

  a) Provide visual feedback using progress bar or steps during processing workflow steps.

  b) Do not provide deep menus. Minimize the depth of menus. Keep simple navigation and minimal page depth.

  c) Provide intuitive UI which is self-explanatory and easy to use.

  d) User should be able to reach the required information with minimal clicks.

  e) Use the points at http://www.w3.org/TR/mobile-bp/ as a checklist for mobile web applications.

3.2.1 Anti Patterns

Given below are some of the common anti-patterns that should be avoided in mobile apps:

- Cluttered information on pages.
- Too heavy and deep page hierarchies with complex navigation.
- Invoking too many services for a given page.
- Not testing mobile apps on all supported devices and platforms.

3.3 CHECKLIST FOR MOBILE APPS

Given below is the checklist that can be used during mobile app development:

- Ensure availability of controls and call to action buttons on all screens.
- Ensure that app handles the crash in a graceful fashion.
- Check for all validations (maximum length, type checking, minimum length etc.) for all forms of input fields.
- Ensure that all kinds of testing is completed to validate the conformance to requirements.
- Check for all alignment, resizing and behavior of UI elements on all supported mobile platforms and browsers.
- Ensure none of the links and screens are broken.
- Ensure that there is no accidental information leakage during exceptions and crashes.
- Ensure that the mobile app is tested for all memory leaks and resource releases once the applications is closed.
- Is the application tested for resource overrun, peak load and other scenarios that lead to app crash?
- Does the app support multi-tasking and multiprocessing?
- Given below are the checklist points from UI point of view:
    - Does the app use standard colors (as per visual specs)?
    - Is the font uniform across pages. Is all text properly aligned?
    - Does text wrap properly around pictures/graphics?
    - Is the error message text spelt correctly on this screen?
    - Does Progress message appear on load of tabbed (active) screens?
- Given below are the checklist points from usability stand point:
    - Verify if the app behaves as desired if device is tilted (portrait/landscape)
    - Verify if the page navigation is smooth
    - Verify if the font size and spacing ensures good readability
    - Verify if the labels and buttons text are clear and concise on every page
    - Verify if the UI elements provide visual feedback when pressed

3.4 USER INTERFACE DESIGN FOR MOBILE APPS

The user interface for a mobile app is one of the critical design item as it plays a major role in user engagement. The main goals of a user interface design for a mobile app are as follows:

- Ease of use: The UI should be easy to learn and easy to use.

- High productivity: User should be able to find the requisite information quickly as well as complete the task quickly
- Easier navigation: User should be able to navigate across the screens easily through intuitive information architecture.
- Minimal error: The app should minimize the error rate for the end user.

3.4.1 Experience Design Process

The key steps in designing the experience (user interface) for mobile apps is given in figure 3.1.

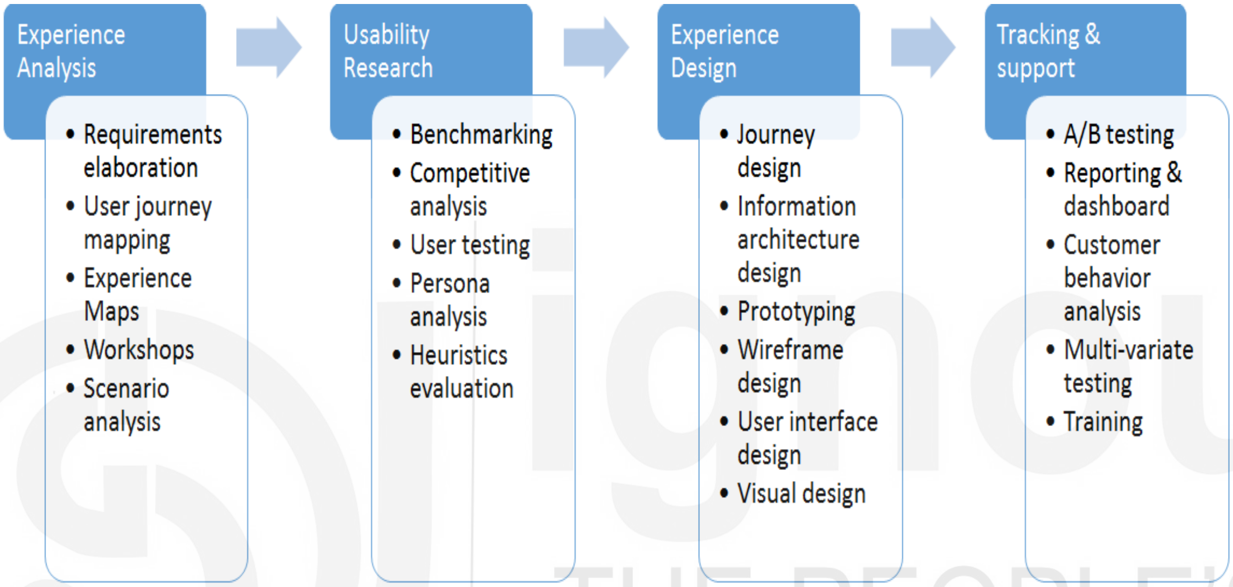

Fig. 3.1: User Interface Design Process for Mobile Apps

Let us look at the key activities in each of the phases:

Experience Analysis phase

In this phase, we would analyze various aspects of mobile app experience. For consumer apps, we always take user-centric approach wherein we place high user engagement and user satisfaction above all other goals. Main activities in this phase are detailed below:

- Requirements elaboration by interviewing key stakeholders and conducting workshops with all users.
- Map the user journey across various user groups and identify key touch points.
- Identify any pain points or challenges in the as-is scenario.
- Develop experience maps for user groups.
- Perform scenario analysis for key user groups.

Usability Research phase

During this phase, we will design for each user journey. Key activities in this phase are as follows:

- Persona analysis: Identify distinct user groups (personas), and map their user journey, tasks, goals and needs.
- Benchmarking and competitive analysis: Benchmark the user design with competitors.
- Heuristics evaluation: Run the key heuristics to ensure that the design conforms to all specified heuristics. Heuristics related to usability, simplicity, user controls, consistency, branding, error handling, flexibility, efficiency, help, aesthetics standards will be tested with help of experts.
- User testing: Test the design with intended users and experts.

Experience Design phase

During design we strive to create simple to use app user interface that enhances various touch points of user journey. Main activities in this phase are given below:

- Wireframe design: We create low-fidelity wireframes (such as sketches, videos) based on journey and persona analysis
- Prototype: Mockups and prototypes are developed to get user feedback. HTML and JavaScript will be used to develop interactive prototypes. Prototype will be tested with stakeholders and users.
- Visual design: Develop the specifications for UI elements such as page layout, fonts, images, buttons, videos and provide design guidelines. Design will be validated through eye-tracking tests.
- Information architecture and navigation model will be defined in this phase. Also, user navigation journey will be defined for various user personas.
- All personalization and contextual features related to mobile app will be designed.

Tracking and Support phase

In this phase, we mainly track the user behavior on the new design to understand the effectiveness of the design. Given below are the key activities.

- A/B testing: We carry out testing with two variants to understand the effectiveness
- Multivariate testing: We test multiple variants of the design.
- Customer behavior analysis: We use analytics to track and analyze the customer behavior.
- Dashboard reporting: We report all the findings and insights in an intuitive dashboard format.

User interfaces of a Mobile app for authentication and dashboard for an insurance company is shown in Figure 3.2.

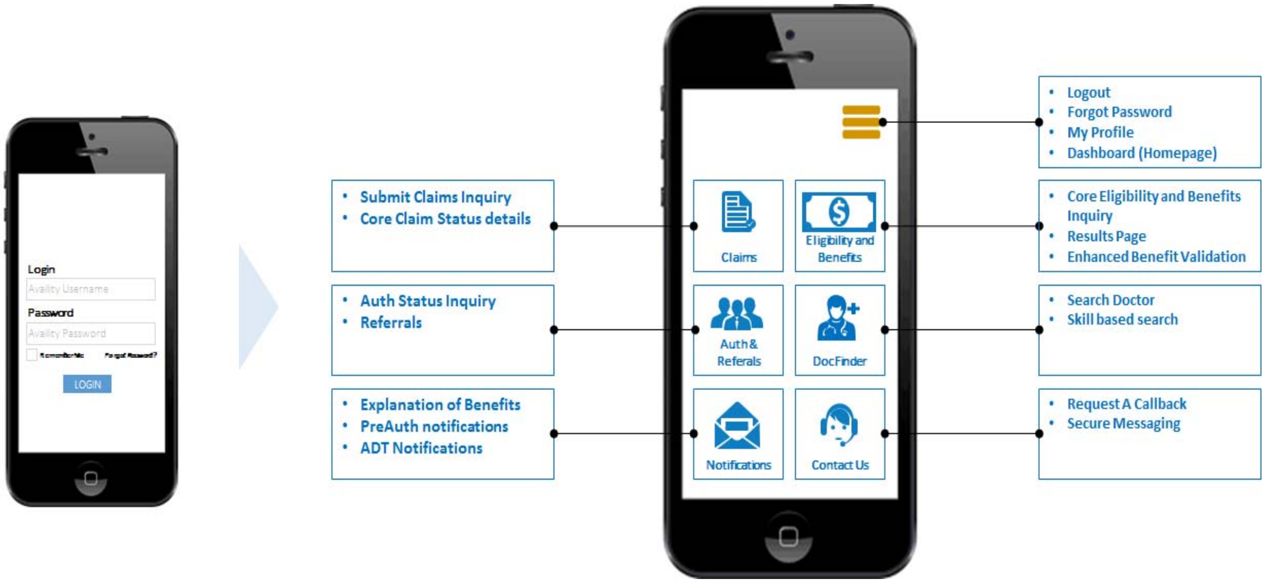

Fig. 3.2: Mobile app user interface of an Insurance Company.

3.5 DEPLOYMENT

Figure 3.3 shows end to end steps involved in the lifecycle of a mobile app deployment.

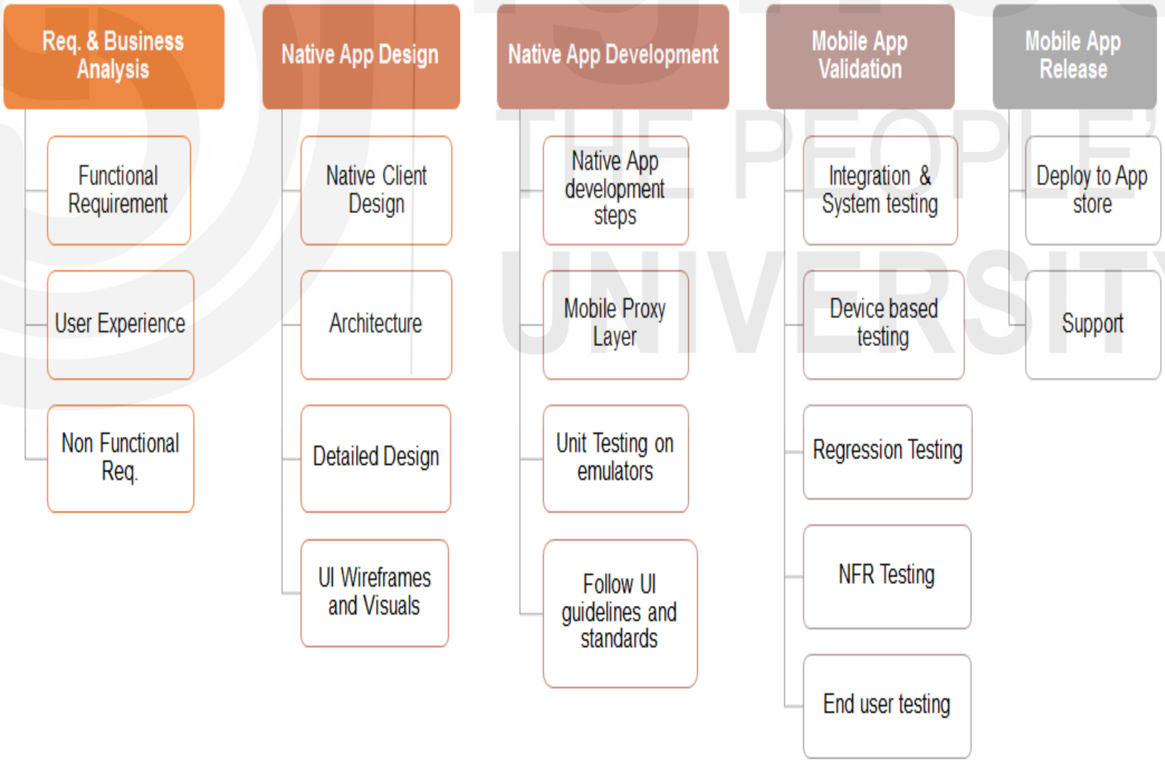

Fig. 3.3: Steps involved in Mobile app deployment

- **Requirements and Business Analysis**: In this phase, we will analyze detailed user requirements, journey mapping, and persona analysis and compile all functional and nonfunctional requirements. We discussed the detailed steps as part of experience design process. We will also understand the requirements related to security, performance, accessibility, localization, standards, modularity, etc.

- **Native App Design:** During this phase, we will design the user experiences, information architecture needed for the mobile app. The end-to-end architecture and detailed design would be carried out for the mobile app. Other key deliverable, in this phase are wireframes, visual design, and HTML prototypes.
- **Native app development:** In this phase, we will follow all the specified design guidelines to develop the native mobile app for the specific platform. Wherever needed we will also develop proxy layer for interaction with other systems and services. The developed app will be tested on various emulators.
- **Mobile app validation:** Various forms of testing such as device testing, integration testing, system testing, security testing, regression testing and NFR (Non-Functional Requirements) testing would be carried out.
- **Mobile App release:** After the testing is completed, mobile app shall be deployed to the corresponding app store and support for future releases shall commence.

☞ **Check Your Progress 1**

1) The process by which we analyze needs of users is known as ……………………………….. .
2) Comparing design with other existing designs is done through ……………………………….. .
3) Developing interactive mockups is done through ………………….. .
4) Comparing two variants of the design is done using ………………. .
5) ……………………………. involves testing app at peak load.

3.6 POWER USAGE

As the battery life for most of the mobile phones is limited, it is important to judiciously use the battery. Hence, mobile apps must be designed and tested for optimal battery usage. Each mobile platform provides its own set of guidelines for optimal power consumption. Given below are some of the generic guidelines for optimal power usage:

- The mobile app should optimally use the CPU
- Usage of disk, Bluetooth and other networks should be minimized
- Disable all unnecessary background services
- Reduce the frequency of app updates
- Regularly monitor the battery usage of all the apps and test the app with peak load

3.7 SYNCHRONIZATION

Mobile apps use synchronization to sync app data with server to refresh the data. A sample sync architecture for iOS mobile app is depicted in figure 3.4

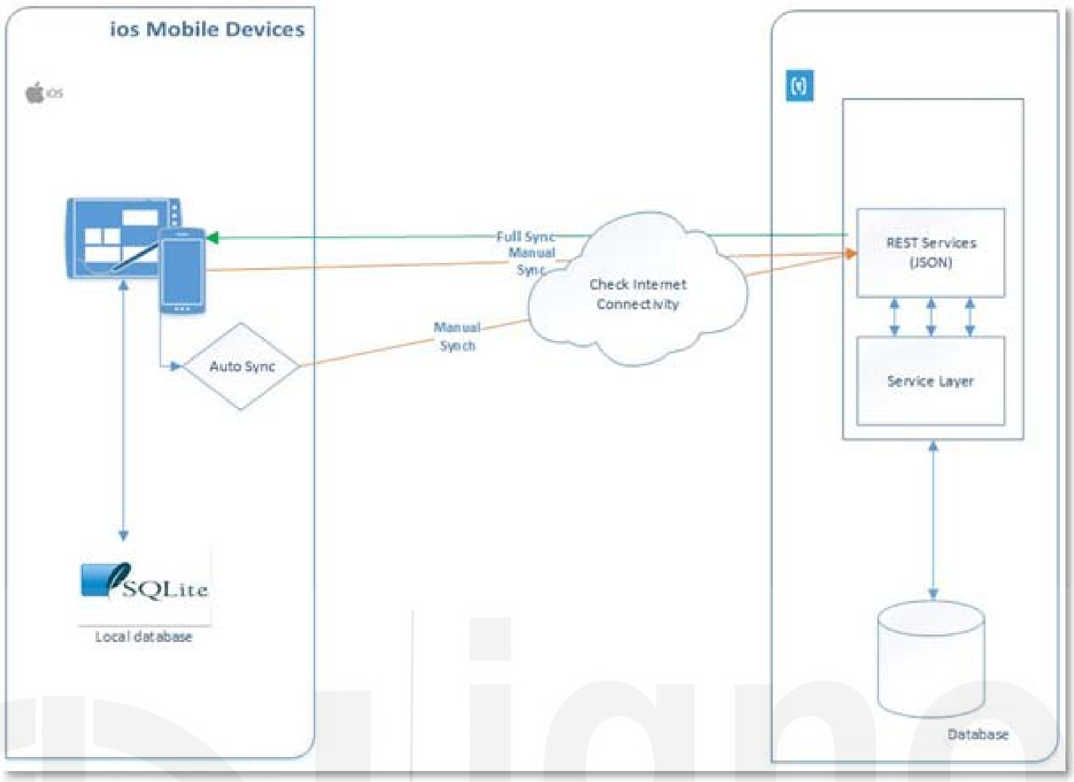

Fig. 3.4: Process of App synchronization

The key steps in the sync process of iOS mobile app are as follows:

i) iOS Offline Mobile application will have two modes of data synchronization:

a) Full Sync

- A full sync involves importing of all the objects. Hence, all the objects in the local store will get refreshed
- Data initialization will happen using full sync upon logging in for the first time

b) Delta Sync

- A delta sync will import only changes and does not import any unchanged record
- Subsequent automatic or manual sync will fetch the data modified after last sync date and time. Offline mobile application will take care of keeping the last sync date on successful sync operation

ii) Data synchronization will happen in both directions from the offline mobile app

- Inbound → for incoming data
- Outbound → for outgoing data

iii) Back end application will expose REST web services to perform Full Sync or Delta Sync

iv) Offline Mobile application will invoke all the required REST web services individually during full or delta sync. This will enable a feature in mobile application of having options for sync of selected entities

v) OAuth (Open Authentication) authentication module will be used for REST based web services.

3.8 PATTERNS AND DESIGN ELEMENTS

Mobile apps are built using platform recommended architecture design patterns to support refactor ability, extensibility and code organization. The following principles of architecture are considered while building mobile apps:

- **Decorator Design Pattern** – Decorator design pattern extensively using Categories, Delegates, and Protocols.
- **Singleton** – Singleton pattern is used for class initiations.
- **Memento** – Memento is used for application state management.
- **Command Pattern** – Command pattern is used for network layer optimization.
- **Façade (Abstract Class)** – Web service API utilization.

In addition to these patterns, following aspects are considered while designing mobile apps:

- **Performance -** Optimized mobile app code and use of best practices will ensure better performance.
- **Security & Standards Compliance –** Security aspects including the following are considered:
    - Sensitive information in platform is encrypted using secure Keychain (AES 256-bit encryption) only
    - Clearing in-memory information at session timeout
    - No Caching of APP data
    - Secure logging
- **Usability -** Allows human interface guidelines, standardized look and feel, navigation, HCI standards etc.
- **Maintainability** - Follows modular approach, platform recommended design patterns, high reusability and extendibility.
- **Code quality** - Strict code reviews and analysis of both static and dynamic nature will ensure high level of code quality.
- **Compatibility -** Follow platform recommended best principles and tools to ensure that app UI is aligned to the recommended form factors to provide completeness.

3.9 SECURITY STANDARDS AND BEST PRACTICES

During mobile application design, it is important to evaluate and take pro-active security measures to mitigate the security risk. The following are some of the practices that will lead to development of a secure Mobile App:

- Security assessment: It is always recommended to assess mobile devices for known security risks and vulnerabilities.

- Security policies:
  - All mobile devices must be password protected.
  - All key applications (such as banking apps) should be password protected.
  - All confidential and user data should be encrypted.
  - All mobile apps should present the privacy policies, data sharing policies, legal policies through end-user license agreements.
  - Password policies should include length restriction (to at least 8 characters), complexity (usage of special characters and alphanumeric), password change frequency and such.
  - All major events such as failed login attempts, apps crashes, and system events should be logged.
  - For secured applications and secure functionality, the system should use multi-factor authentication or mobile device management (MDM) capability.
  - Data at rest and in motion should be encrypted using appropriate encryption standards.
- Various Authentication mechanisms for mobile apps:
  - **Single factor authentication:** Here, user is asked to enter a password every time the application is started or any secured activity is being initiated.
  - **Two factor authentication:** The authentication is performed twice, once with the user credentials and second using the OTP (One time password).
  - **Single Sign on (SSO):** Mobile app is integrated with enterprise SSO solutions to seamlessly access all secured enterprise applications.
  - **Other modes of authentication:**
    - Authentication using popular social channels like Google+, Facebook, Twitter, etc.
    - Authentication using Biometrics for e.g. facial features, speech patterns, fingerprints, etc.
- The following Information Risk Management (IRM) policies should be applied:
  - Periodic threat and vulnerability assessment of all the applications.
  - Remote data wipe methods should be enabled for administrators.
  - Mobile device disposal policies should be devised and enforced to protect the confidential and business critical data.
  - Virus and malware scan should be carried out on periodic basis.
  - Filters and scanners should be installed to prevent vulnerabilities such as phishing, data leakage, cross-site scripting, etc.
  - Screen locking policies should be enforced.
  - Restrict the installation of applications such as Jailbreak to prevent unauthorized usage.

3.9.1 Mobile Platform Security

The following are the security related practices for securing mobile platform:

- **Data Transmission Security** - Provide secure connection for end to end mobile communications.
- **Operational Data Security** - Minimize exposure of data across all end points.
- **Communication Channel Security** - Provide secure communication channels for transmitting confidential data.
- **Application Security** - Provide role based access to all functionality and data and provide access only for authorized roles.
- **On-Device Data Security** - Encrypt data that resides on the device for native and hybrid apps.
- **Encryption Standards**: Use encryption standards such as SHA2 (Secure Hash Algorithm) to encrypt sensitive information.
- All mobile apps should use filters, validations and other secure mechanisms to address following vulnerabilities:
  - Invalidated input
  - Broken access control
  - Broken authentication and session management
  - Cross site scripting (XSS) flaws
  - Buffer overflows
  - Injection flaws (e.g., SQL injection)
  - Improper error handling
  - Data under-run / overrun
  - Application denial of service
  - Insecure configuration management
  - Improper application session termination
  - Insecure storage and transmission
  - Insecure configuration management
  - Viewing instructions or code in the server script
  - Modification by web page users
  - User-entered input used for script code injection
  - Access via other non-web-based services
  - Dynamic generation of other server-side scripts
  - Dynamically generating executable content (beyond HTML)
  - Not running as a user ID with least privilege (Running with system level privilege)
  - Running in a system shell context
- Use secure transport layer (SSL/HTTPS) for secured data transmission.

- App should not log any sensitive data.

  The following practices may be followed for source code security:

  - Regular security review of code.
  - Secured access controlled source code repository.
  - Proper version management and release management processes.

3.10 MOBILE APP TESTING

As testing is an important aspect to ensure quality of the mobile apps, let us look at various testing scenarios for mobile apps. Mobile testing is challenging considering the variety of form factors, hardware, unreliable wireless network, latency issues, etc. Given below are some of the key testing categories that are carried out for mobile app testing.

3.10.1 UI (User Interface) Testing

In UI testing category, we would test the user experience for various form factors, resolutions and on various browsers. The testing normally includes testing look and feel, font, color, controls, touch controls, pinch controls, ease of use, control features, zoom in and zoom out features, etc.

3.10.2 Unit Testing

In this testing category the logical unit of testing would be a module or a screen. We would test the module functionality, screen features, navigation features, flows, business rules as part of it.

3.10.3 Integration Testing

This testing is carried out with integrated set of modules. We will mainly test the data flow, performance, dependencies.

3.10.4 System Testing

System testing involves end-to-end testing for the entire system. This includes testing all the scenarios end to end to check if the application meets the requirements.

3.10.5 Compatibility Testing

This includes testing the mobile app on all supported devices and hardware platforms. We would also test the mobile app for various networks, browsers and carriers. Leverage automated tools to test on various combinations of form factors, devices and mobile platforms.

3.10.6 Performance Testing

The mobile app will be tested at various loads and various bandwidths for response times. Identifying bottlenecks is one of the key activities in this phase of testing. The resource usage such as battery usage will be tested along with scalability and reliability.

3.10.7 Security Testing

This includes testing various security scenarios of authentication, authorization, authorized port access, checking user permissions check, etc. We will also verify potential vulnerabilities such as SQL injection, input validation, account lockout scenarios, session handling, communication, data encryption, information leakage, etc.

3.10.8 Synchronization Testing

This testing involves testing synchronization scenarios between mobile device and the server. This also includes testing data integrity during synchronization process and handling network failure scenarios.

3.10.9 Usability Testing

The testing is carried out from end user stand point of view to test the ease of use, efficiency, recall, ease of navigation, etc.

In addition to the above mentioned tests, we would also conduct other tests based on the application needs:

- Installation testing to test the ease of installation on various platforms
- Recovery testing to understand the ease with which the app recovers from failure.
- Battery consumption testing to test the usage of battery for extended durations
- Network testing to check the app behavior on various networks with differing bandwidths.
- Interruption testing to check the behavior of app during interrupts such as incoming call, message, flash message, system events etc.
- Compatibility testing is carried out mainly for mobile web apps to test the experience and behavior on various devices and mobile platforms.
- Localization testing to test the app behavior for various languages, translation needs and cultural requirements.

☞ Check Your Progress 2

1) For optimal power usage, frequency of app updates should be ……………. and unnecessary background services should be …….. .

2) …………………… pattern is mainly used for class initiations.

3) ……………………… design concern deals with look and feel and interface guidelines.

4) ………………………………………… authentication forces users to authenticate multiple times.

5) …………………………… involves end-to-end testing for the entire system.

3.11 SUMMARY

In this unit, we started discussing the key design considerations, anti-patterns and best practices in mobile app design. We provided a checklist and then provided deep dive concepts of user interface design for mobile apps. We also looked at the power usage and synchronization aspects of mobile app design. We then looked at various patterns, security standards and various forms of mobile app testing.

3.12 ANSWERS TO CHECK YOUR PROGRESS

Check Your Progress 1

1) Persona analysis
2) Benchmarking and competitive analysis
3) Prototyping
4) A/B testing
5) Regression testing

Check Your Progress 2

1) Minimized and disabled
2) Singleton
3) Usability
4) Two factor
5) System testing

3.13 FURTHER READINGS

References

- https://en.wikipedia.org/wiki/Mobile_application_testing
- https://en.wikipedia.org/wiki/User_interface_design
- https://www.owasp.org/index.php/OWASP_Mobile_Security_Project