Chapter 18

- **MobileApp Design**

Slide Set to accompany

Software Engineering: A Practitioner's Approach, 8/e

by Roger S. Pressman and Bruce R. Maxim

Mobile Development Considerations – 1

- Multiple hardware and software platforms
- Many development frameworks and programming languages.
- Many app stores with differing acceptance rules and tool requirements
- Short development cycles
- User interface limitations

Mobile Development Considerations – 2

- Complex camera/sensor interaction
- Effective use of context
- Power management
- Security and privacy models/policies
- Device limitations (computation and storage)
- Integration of external services
- Texting complexities

MobileApp Development Process Model

- Formulation
- Planning
- Analysis
- Engineering
- Implementation and testing
- User evaluation

MobileApp Quality Checklist - 1

- Can content and/or function and/or navigation options be tailored to the user's preferences?
- Can content and/or functionality be customized to the bandwidth at which the user communicates? Does the app account for weak or lost signal in an acceptable manner?
- Can content and/or function and/or navigation options be made context aware according to the user's preferences?
- Has adequate consideration been given to the power availability on the target device(s)?
- Have graphics, media (audio, video), and other web or cloud services been used appropriately?

MobileApp Quality Checklist - 2

- Is the overall page design easy to read and navigate?
- Does the app take screen size differences into account?
- Does the user interface conform to the display and interaction standards adopted for the targeted mobile device(s)?
- Does the app conform to the reliability, security, and privacy expectations of its users?
- What provisions have been made to ensure app remains current?
- Has the MobileApp been tested in all targeted user environments and for all targeted devices?

MobileApp User Interface Quality Requirements

- Is the overall page design easy to read and navigate?
- Does the app take screen size differences into account?
- Does the user interface conform to the display and interaction standards adopted for the targeted mobile device(s)?
- Does the app conform to the reliability, security, and privacy expectations of its users?
- What provisions have been made to ensure app remains current?
- Has the MobileApp been tested in all targeted user environments and for all targeted devices?

MobileApp User Interface Design Considerations

- Define user interface brand signatures
- Focus the portfolio of products
- Identify core user stories
- Optimize UI flows and elements
- Define scaling rules
- Create user performance dashboard
- Rely on dedicated champion with user interface engineering skills

MobileApp Design Mistakes

- Kitchen sink
- Inconsistency
- Overdesigning
- Lack of speed
- Verbiage
- Non-standard interaction
- Help-and –FAQ-itis

MobileApp Design Best Practices

- Identify the audience
- Design for context of use
- Recognize line between simplicity is not laziness
- Use the platform to its advantage
- Allow for discoverability of advanced functionality
- Use clear and consistent labels
- Cleaver icons should never be developed at the expense of user understanding
- Long scrolling forms trump multiple screens

Assessing Mobile Interactive Development Environments

- General productivity features
- Third-party SDK integration
- Post-compilation tooks
- Over the air development support
- End-to-end mobile application development
- Documentation and tutorials
- Graphical user interface builders

MobileApp Middleware

- Facilitates communication and coordination of distributed components
- Allows developers to rely on abstractions and hide mobile environment details
- Helps MobileApps to achieve context awareness as required

Service Computing

- Focuses on architectural design and enables application development through service discovery and composition
- Allows MobileApp developers to avoid the need to integrate service source code into the client running on a mobile device
- Runs out of the provider's server
- Loosely coupled with applications
- Provides an API to allow service to be treated like an abstract black box

Cloud Computing

- Focuses on the effective delivery of services to users through flexible and scalable resource virtualization and loading balancing
- Lets the client (either a user or program) request computing capabilities as needed, across network boundaries anywhere or any time

Cloud Computing Architecture

- Cloud architecture has three service layers
  - **Software as service** layer consists of software components and applications hosted by third-party service providers
  - **Platform as service** layer provides a collaborative development platform to assist with design, implementation, and testing by geographically distributed team members
  - **Infrastructure and service** provides virtual computing resources (storage, processing power, network connectivity) on the cloud