

Logical Operations Master Mobile Application Developer (MMAD) Exam MAD-111

Exam Information

Candidate Eligibility:

The *Logical Operations Master Mobile Application Developer (MMAD)* exam requires no application fee, supporting documentation, or other eligibility verification measures for you to be eligible to take the exam. Simply purchase an exam voucher [here](#), then Logical Operations will send you an email containing the information you need to register to take the exam through Pearson VUE. You can also purchase a voucher directly through Pearson VUE. If your voucher came bundled with your MMAD training program, you will receive registration information from your trainer or training administrator. Once you have obtained your voucher information, you can register for an exam time [here](#). By redeeming your exam voucher, you agree to our [Candidate Agreement](#).

Exam Prerequisites

While there are no formal prerequisites for taking the MMAD exam, Logical Operations strongly recommends you have training and practical experience developing mobile applications in a variety of programming environments and that you can implement countermeasures to common mobile application security threats. You should have some experience with the following:

- Developing applications using object-oriented programming languages, such as Java, C++, C#, and Visual Basic
- Identifying the security vulnerabilities in both the Android and iOS system architectures
- Employing strategies to promote the security of mobile apps
- Hardening mobile apps against attack at levels appropriate to the corresponding risk model

Logical Operations also recommends that you have basic project management skills specifically aligned to projects involving mobile application development. You can obtain this level of skill and knowledge by taking the following Logical Operations courses:

- *Developing Secure Android™ Apps for Business*
- *Developing Secure iOS® Apps for Business*
- *Project Management Fundamentals (Third Edition)*

Exam Specifications

Number of Items: 75

Passing Score: 57 out of 75 (75%)

Duration: 120 minutes (**Note:** Published exam times include the 5 minutes you are allotted for reading and signing the Candidate Agreement and the 5 minutes you are allotted for the Pearson VUE testing system tutorial.)

Exam Options: In person at Pearson VUE test centers

Item Formats: Multiple Choice/Multiple Response/True-False

Exam Description

Target Candidate:

This exam is intended for an application programmer or web developer who is relatively new to developing Android and iOS apps. The candidate has experience using a computer running Windows 10 or Mac OS X and developing applications using object-oriented programming languages and tools. The ideal candidate also has basic project management skills and experience.

To ensure exam candidates possess the aforementioned knowledge, skills, and abilities, the MMAD exam will test them on the following objective domains with the following weightings:

Domain	% of Examination
1.0 Android Mobile Application Development	40%
2.0 iOS Mobile Application Development	38%
3.0 General Principles of Secure Mobile Application Development	13%
4.0 Project Management Fundamentals	9%
Total	100%

The information that follows is meant to help you prepare for your Logical Operations certification exam. This information does not represent an exhaustive list of all the concepts and skills that you may be tested on during your exam. The exam domains, identified previously and included in the objectives listing, represent the large content areas covered in the exam. The objectives within those domains represent the specific tasks associated with the job role being tested. The information beyond the domains and objectives is meant to provide examples of the types of concepts, tools, skills, and abilities that relate to the corresponding domains and objectives. All of this information represents the industry-expert analysis of the job role related to the certification and does not necessarily correlate one-to-one with the content covered in your training program or on your exam. Logical Operations strongly recommends that you independently study to familiarize yourself with any concept identified here that was not explicitly covered in your training program or products.

Objectives:

Domain 1.0 Android Mobile Application Development

Objective 1.1 Given a set of requirements and expectations, develop an Android app with multiple activities

- Android operating system
 - Versions
 - Marshmallow
 - Nougat
 - Version features
 - Hardware support
 - API levels
- Android app development tools
 - JDK
 - Android Studio
 - Android SDK
 - Android SDK Manager
 - Command-line tools
 - Android Virtual Device Manager
 - Emulators
- Android architecture
 - Kernel
 - Libraries
 - Android Runtime
 - Application framework
 - Applications
- The Android project structure
 - Manifests
 - Java
 - Res
- Activities
 - Layouts
 - View groups and objects
 - Intents and bundles
 - Resource files and directories
 - Fragments and multiple activities
- The app lifecycle
 - App organization
 - Device screens
 - Device and app buttons

- UI states
- The activity lifecycle
- The fragment lifecycle
- Memory management and multitasking
- Design requirements and expectations
 - Material design
 - Android user expectations
 - Mobile device constraints

Objective 1.2 Implement the use of resources, multimedia assets, data, and storage

- Resources
 - Resource loading
 - Languages
- Themes and styles
- Images
- Code-generated graphics
- Animation
- Data storage
 - Data storage preferences
 - Internal storage
 - External storage
 - SQLite
 - Remote storage
- Write data
- Read data
- Web data
 - AsyncTask
 - XML data
 - ListView adapter
 - Predefined list layouts
 - WebView

Objective 1.3 Perform Android app debugging and analysis tasks

- Android Debug Bridge
 - Frames
 - Threads
 - Variables
 - Watches
- Breakpoints
- LogCat
- Android Device Monitor
 - DDMS

- Dev Tool app

Objective 1.4 Implement a user preferences screen to enable customization

- Preferences layout
- Activity and fragment preferences
- Preference management

Objective 1.5 Integrate app functionality with system resources

- Alternate layouts
 - Screen size
 - Screen layout
 - Pixel density
 - API versions
- Sensors
 - Motion
 - Environment
 - Position
- Sensor framework
- Location services
- Multimedia capture
- Multimedia framework
- Widgets

Objective 1.6 Finalize an Android app

- Android mobile app security
 - Linux kernel security
 - App isolation
 - APK files
 - Permissions
 - Installation
 - Runtime
 - Encryption/decryption
 - JCA
- Gradle and manifest preparation
- Method reference limit
- ProGuard
- Digital signatures

Domain 2.0 iOS Mobile Application Development

Objective 2.1 Understand iOS app design standards

- Design process phases
 - Define the concept
 - Identify the data sources
 - UI design

- Define the interactions
- UI standards
- iOS user expectations
- iOS design patterns
 - MVC
 - Target-Action
 - Delegation

Objective 2.2 Given a set of requirements and expectations, develop an iOS mobile app

- iOS development tools
 - Xcode
 - Features
 - Preferences
 - Navigation
 - The Simulator application
 - The iOS SDK
- Cross-platform tools
 - Sencha
 - PhoneGap
 - Appcelerator Titanium
 - Qt
 - Xamarin
 - Alpha Anywhere
 - Microsoft Visual Studio
- Apple developer program accounts
- Swift
 - Swift Standard Library
 - The playground
 - Comments
 - Variables
 - Constants
 - Data types
 - Named
 - Compound
 - Optional
 - Custom
 - Operators
 - Assignment
 - Arithmetic
 - Comparison
 - Logical

- Access levels
 - Public
 - Internal
 - Private
 - Flow control
- iOS frameworks
 - Core OS
 - Core Services
 - Cocoa Touch
 - Media
- iOS app projects
 - Project templates
- UI layouts
 - Views
 - NIBs and XIBs
 - The Storyboard Editor
 - Constraints
- View events and user interactions
- Actions and outlets

Objective 2.3 Implement multiple view navigation

- Navigation patterns
 - Single view
 - Linear
 - Hierarchical
 - Hybrid
- Navigation components
 - Navigation controller
 - Tab bar controller
 - Page view controller
 - Custom classes

Objective 2.4 Implement the use of data, graphics, and media

- Data storage options
 - Data storage preferences
 - Property lists
 - Embedded resources and files
 - Local app files
 - JSON
 - SQLite
 - Core Data
 - Web services

- Cloud storage
 - Keychain services
- The iOS file system
- Table view
- Preferences view
- Segues
- OpenURL
- UIImage view
- Core Graphics
 - UIView Drawing model
- iOS Animation
 - Core Animation

Objective 2.5 Integrate app functionality with system resources

- App states
 - Background state
- State change methods
- View controller methods
 - Stack view
 - On-screen keyboard
- Queued notifications
- Capabilities
 - In-app purchase
 - Push notifications
 - Maps
 - Map Kit Framework
 - Core Location

Objective 2.6 Perform debugging and maintenance tasks

- Xcode debugging features
- Errors
 - Simple domain
 - Recoverable
 - Logic failure
 - Universal
- Error reporting and handling
- Alerts

Objective 2.7 Finalize an iOS app

- iOS security architecture
 - Software
 - Hardware and firmware

- iOS security features
 - Secure boot chain
 - System software authorization
 - Secure enclave
 - Touch ID
 - Encryption
 - File data protection
 - Network security
- App Store review
- The iOS Keychain API
- Internationalization and localization
- App release

Domain 3.0 General Principles of Mobile Application Security

Objective 3.1 Explain the need for mobile application security, and interpret security requirements and expectations

- Consequences of lax security
- Information asset threats
 - Hacking
 - Untrusted networks
- Development standards
 - Platform requirements
 - Google Play
 - Apple App Store
 - Organizational processes and policies
 - Least privilege
 - Permissions

Objective 3.2 Compare and contrast the Android and iOS security architectures

- Platform strengths and weaknesses
 - iOS architecture
 - Layers
 - iOS security framework
 - Hardware/firmware
 - File system
 - Android architecture
 - Linux kernel security
 - Permission model
 - Android/iOS vulnerabilities
 - Rooting/jailbreaking
 - Developer-caused
 - Third-party/unauthorized apps

Objective 3.3 Implement mobile application development security measures

- Security best practices
- Authentication and authorization
- Session management
- Common threats
 - SQL injection
 - Buffer overflows
- Input/output handling
- Exceptions
- Local device and process access
 - IPC
- Data encryption
 - Encryption approaches
 - Hashing
 - JCA
 - Crypto Engine certificates
 - Password salting
- Local storage access
 - Local storage types
 - File-based storage
 - User preferences
 - Keychain storage
 - SQLite databases
 - Directory structure
 - Threats to stored data
 - Storage permissions
- Network and web communication security
 - Input validation
 - Secure network communication
- WebView/UIWebView component use
 - Java script injection
- Credential protection
 - User authentication
- App hardening techniques
 - Reverse/forward engineering
 - Static/dynamic analysis
 - Data/code/stack trace obfuscation
 - Password storage
 - Function naming

Domain 4.0 Project Management Fundamentals

Objective 4.1 Identify basic project management concept

- Projects, programs, and portfolios
- Stakeholders
 - Sponsors
 - Customers
 - Project managers
 - Team members
- Deliverables
- The project management process groups
- Roles and responsibilities
- Constraint factors
 - Time
 - Cost
 - Scope
- Scope creep
- Project objectives
- Project assumptions
- Risk

Objective 4.2 Define project requirements and create a project plan

- WBS
 - Decomposition
- Activities/sequencing
- Dependencies
- Precedence relationships
- Resources
 - Estimation
 - Leveling
- The critical path
 - Float
- Baselines
- Costs
 - Estimates
 - Budgeting
 - Cost baselines
- Risk analysis
- Communication plans
- Change control

Objective 4.3 Execute an app development plan

- Project work

- Quality assurance
- Tracking project progress
 - EVA
 - Variance
 - Performance reporting
- Project closeout
 - Closeout elements
 - Final reports

Continuing Education Requirements

The *Logical Operations Master Mobile Application Developer* (MMAD) certification is valid for 3 years from the time the certification is granted. You must re-take the most up-to-date version of the exam prior to the 3-year period's end to maintain a continuously valid certification.

To view the Logical Operations Candidate Agreement, click [here](#).

Then purchase a voucher to take the exam by clicking [here](#).