

HCL Volt MX 9.2

Certified Associate

Developer Project Packet



Table of Contents

- What is the HCL Volt MX Certified Associate Developer Badge? 3
- How do I earn the HCL Volt MX Certified Associate Developer Badge? 3
- What are the requirements for the HCL Volt MX Certified Associate Developer Badge?3
 - UI and Application Requirements 4
 - UI Requirements 4
 - Application Requirements 5
- Scoring..... 5
- Associate Developer Project Showcase 5
- Disclaimers 6
- Certification Review Process 7
- HCL Volt MX 9.2 Certified Associate Developer Project Presentation Checklist 8
- HCL Volt MX 9.2 Certified Associate Developer Project Rubric 9

What is the HCL Volt MX Certified Associate Developer Badge?

The HCL Volt MX Certified Associate Developer Badge shows that you can build both Mobile and Web applications that illustrate a concrete understanding and key execution of product development principles. Creation of these applications will include reading and writing data using HCL Volt MX Iris & Foundry.

How do I earn the HCL Volt MX Certified Associate Developer Badge?

The HCL Volt MX Certified Associate Developer Badge is earned when an applicant creates a project that is evaluated by the Digital Solutions Academy that passes all requirements as specified in the [HCL Volt MX Certified Associate Developer Rubric](#).

To aid in your background education and understanding of HCL Volt MX, it is recommended that you complete the self-paced or onsite delivery of the HCL Volt MX Bootcamp. This training is optional and not required to achieve the Associate Developer badge.

What are the requirements for the HCL Volt MX Certified Associate Developer Badge?

Applicants will be required to develop both a Mobile and Web application that contains key functionality requirements that demonstrate your working knowledge and skills using HCL Volt MX Iris and Foundry. The Mobile application will allow you to select data from one source and save it to a different data source. The Web application will show an aggregate view of the data you've saved.

Data Requirements

To create your mobile and web applications, first think of a Business Industry that would benefit from what you will create. Build a use case for how the application will fulfill a need for the selected industry.

Here are a few Industry ideas and Use Cases to get you thinking:

Business Industries (non-conclusive)

Healthcare

Insurance

Hospitality Industry (Hotels and Motels)

Construction

Automobile Manufacturing

Energy

Retail

Business Use Cases

Workflow Approval Process

Inspection

Event Management

The subject of your app is your choice. Where you get the data that drives your app is your choice. Select data that you can build around your use case and business industry. Create an application's use case that not only illustrates your knowledge of building applications, but one that may be useful or valuable to others.

Be sure that the data supports the search and filter requirements for a compelling application. It is also recommended that you find data that can provide image or geo-location. This will allow you to incorporate the Camera and/or Map widgets.

Example Data Sources:

<https://github.com/public-apis/public-apis>

**Note: The Digital Solutions Academy nor HCL Software neither support nor maintain the public apis provided by this source.*

UI and Application Requirements

You should create a compelling use case that includes the ability for the Mobile and Web applications to work together seamlessly.

For example, if your Mobile application supports a “shopping cart”, “purchases” must be visible in the Web app. Data saved in the Mobile app should be reflected in the Web app, and vice-versa. Your apps must read data from an externally available data source and save data using Foundry Storage Objects.

Full requirements can be found in the [HCL Volt MX Certified Associate Developer Rubric](#).

UI Requirements

Your Mobile App must include at least 3 screens; a Login, Home, and at least one more screen of your choosing based on your use case. It must have a way to search for data and display the results and the details in an organized and compelling fashion.

Your Web Application must incorporate the same Login and Home Screen concepts in a Responsive format. This means that it must redraw differently based on the screen size.

Here are some details on the Login and Home screen requirements

Login screen

- It must have “Save my username” functionality
- Include your company logo AND a background image
- All widgets in this form should not use the out-of-the-box colors
- Use the Foundry User Store Identity provider for authentication
- Capture profile information after login to display the current logged in user on your other screens

Home screen

- This should be the main screen after you've logged in
- Logged in user should be clearly displayed
- Include options to search for data and navigate between screens
- All screens should have a common header and footer created as a component without contract
- This screen should also represent your color theme

Application Requirements

The application must demonstrate your ability to write JavaScript code integration with the Volt MX Iris and Foundry APIs using best practices. You must use both “Low” Code and “Pro” Code to demonstrate the skills to use both methods in your project.

Coding Standards

- **Your code must be clean, consistent, and understandable for Academy review**
- Do not keep the default names for your widgets
- Use the Invoke Function action to call methods you’ve written in controller files or global JavaScript modules
- Limit your use of snippets
- If you include any external libraries (even your own), please let us know

Limit excess data calls

- Determine when data should be cached and when it shouldn’t so that current data is shown when navigating between screens
- Support data flow from one form to the other (and back)
- Present how you are caching data or manipulating and be prepared to review the code you’ve written to support this

Device and Responsive Web Requirements

- Support both Android and iPhone
 - The mobile app should look as close as possible on both Android and iPhone
 - You can choose iPhone SPA or Android SPA or a Tablet as your second choice, but you must be able to run your application on at least one device using HCL Volt MX App Preview
- Your Responsive Web app must demonstrate a different layout for at least 2 screen sizes

Scoring

Candidate projects will be scored against the core areas including mobile application, web application, Foundry, and UI requirements. A maximum score across these core areas is **40** Points. Candidates scoring **35** points or above will earn the HCL Volt MX Certified Associate Developer Badge.

Associate Developer Project Showcase

Those candidates who supply especially captivating and powerful mobile and web applications that clearly demonstrate their development skills and the capabilities of HCL Volt MX product will be selected to receive special recognition in the DS Academy Project Showcase.

If you are selected, we will include your photo, bio, project details on the Academy website! We will also promote your project as a demo to customers and partners seeking to understand features, functions, and capabilities across industries.

Disclaimers

According to codequiry.com, source code plagiarism is defined as copying or reproducing source code without written permission from the original creator. That includes adapting the code minimally, moderately or including fragments of the original author's code in your own code.

We understand that in the “real world”, there are many online sources (Google, Stack Overflow), colleagues, co-workers, and friends that can provide you with assistance while developing code for projects and applications.

However, you are agreeing that when you submit your project to earn the HCL Volt MX Certified Associate Developer badge, that:

- you completed the work yourself,
- you did not copy someone else's code or fragments thereof,
- someone else did not do the work on your behalf.

Certification Review Process

- Develop
 - Using the [HCL Volt MX 9.2 Certified Associate Developer Project Rubric](#) and additional requirements in this packet, build your mobile and web applications. Ensure that they include all requirements as specified.
- Prepare
 - When you have completed your applications, prepare to present your project to an HCL Volt MX expert via the DS Academy using the [HCL Volt MX 9.2 Certified Associate Developer Project Presentation Checklist](#).
- Schedule
 - Request to schedule your “*HCL Volt MX 9.2 Certified Associate Developer Certification Review*” by [filling out the form](#).
- Deliver
 - Once you have received a meeting invitation from the Academy, show our experts what you’ve developed, following the [Presentation Checklist!](#)
- Results
 - After your scheduled presentation, the Academy will review your scored and inform you of your results within 3-5 business days.

HCL Volt MX 9.2

Certified Associate Developer Project Presentation Checklist

- Working Mobile Application
- Working Web Application
- Screenshots or Brief Video showing the Applications' capabilities
- Presentation that includes:
 - Targeted Industry
 - Use Case / User Story
 - Live Demo of the Mobile Application (Showcasing Rubric Specs)
 - Live Demo of the Web Application (Showcasing Rubric Specs)
- Project Source Code & Files

Important Notes:

- The Academy will not review or validate projects that do not work.
- If you fail on the second submission, you will need to present your certification with a new backend data source and UI.

HCL Volt MX 9.2 Certified

Associate Developer Project Rubric



Design Evaluations	4	3	Score
1 - Overall UI Design	The overall UI, color scheme, and design was exceptionally attractive.	The overall UI, color scheme, and design was attractive and easy to use.	
2 - Overall UX Experience	The overall application flow was extremely intuitive and very easy to use.	The overall application flow was intuitive and easy to use.	
3 - Overall Developer Experience	Proper coding techniques and naming conventions were used.		
User Interface Requirements	4	3	Score
4 -Login Screen	The application includes the Login Screen and all listed requirements.	The application includes the Login Screen and 3 requirements.	
2 or more colors			
Company & Background Logo			
Save Me functionality using local data store API			
User Repository Identity Service			
5 -Home Screen	The application includes the Home Screen and all listed requirements.	The application includes the Home Screen and 3 requirements.	
Display Profile in Home Screen			
Represents the same color theme			
Looks same in Android and iPhone			
Common Header/ Footer with Component w/o Contract			

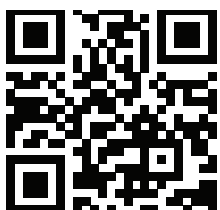
UI Additional Requirements	4	3	Score
6-Additional Screens	The application includes 6 of the listed requirements.	The application includes 4 of listed requirements.	
Responsive Web Design – Design your screen so it shows different layouts in different sizes			
Uses Segments with Sections			
Uses Segments with Templates or Dynamic Skinning			
Includes Dependent Dropdowns/Lazy Loading Must use selection from one of the selection widgets to filter data shown in another selection widget			
Results cached for navigation			
Additional data shown in “Detail View” when compared to “List View”			
Common Header/Footer with Component w/o Contract			

Iris Widgets Requirements	4	3	Score
7- Iris Widgets	The application includes at least 10 Iris Widgets. Student to indicate which Screen contains the Widget.	The application includes 6 Iris Widgets. Studentto indicate whic Screen contains the Widget.	
Any Text Widget - Field Validation must be done on data entry for at least one widget			
Button			
Label			
Image - Display different images based on resolution or device type			
Flex/FlexScroll Containers - Use Flow Horizontal or Flow Vertical layout			
Selection Widget (List/Check/Radio)			
Segment - Make use of the widgetDataMap			
Templates - Use templates to display differently on a row-by-row or section basis			
Segments with Sections - Display data using sections, including a section header			
Browser			
Calendar - You must demonstrate date range validation			
Camera - Display a photo in an image widget			
Map - Must use custom callouts and/or a custom pin			

Applications Requirements	4	3	Score
8- Application Items	The application includes 8 of the Application Items. Student to indicate which Screen or Foundry service supports this functionality.	The application includes 6 Application Items. Student to indicate which Screen or Foundry service supports this functionality.	
CRUD: Create and Update data using Storage Objects			
CRUD: Use ODATA to aggregate or filter data			
Data Caching – Data is cached during navigation to reduce service calls			
Common Navigation across all device types, using the back button in the header component			
Skins / Themes Custom skins must be properly renamed			
Local Data Store – Save and retrieve data from the local data store			
Geolocation API			
Components-Must user at least one component in more than one screen			
Collections – Show the use of collections within your project. You can demonstrate how you have created and used collections during your			
Application Timeout – after a configurable period of activity logout and return user to login screen			

Integration Requirements	4	3	Score
9- Foundry Requirements	This application includes all functionality.		
Identity Service Your application must contain an Identity Service and work with at least 2 sample users			
Integration Your application must support using a public API and filtering of data			
Storage Objects Your application must support saving and filtering data using local storage objects in an online mode			

Additional Features	4	3	Score
<p>10 - Wildcard Features (Add these items to achieve more points) This section helps you to implement additional features in your application, for additional points or to account for missing points in the rubric</p>	The application includes 6 or more Wildcard Features.	The application includes 4 Wildcard Features.	
<p>Orchestration Connect 2 services in concurrent or sequential mode and use this in your application</p>			
<p>JavaScript Pre or Post Processor Create a processor with JavaScript to either validate data or manage the data before returning it to the device</p>			
<p>Component With Contract Create a Component with Contract that is used in more than one screen in your application</p>			
<p>Animation Include animation in your project</p>			
<p>Include NFI For example, display device info on a Settings page</p>			
<p>Include a Splash Screen in your mobile application</p>			
<p>Create a Custom or Java Adapter</p>			
<p>Implement Segment Row or Column sorting</p>			
<p>Include any type of Charting</p>			
<p>Include Gestures in your mobile application</p>			
<p>Use a Component from the Forge Marketplace</p>			
<p>Localize your app in more than one language</p>			



About HCL Software

HCL Software, a division of HCL Technologies (HCL) develops, markets, sells, and supports over 30 product families in the areas of Customer Experience, Digital Solutions, DevSecOps, and Security and Automation. HCL Software is the cloud native solution factory for enterprise software and powers millions of apps at more than 20,000 organizations, including over half of the Fortune 1000 and Global 2000 companies. HCL Software’s mission is to drive ultimate customer success with its IT investments through relentless product innovation. please visit www.hcltechsw.com