Player Registration				
App ID	provide value			
Client ID	provide value			
VCID	provide value			
Ad Measurement	provide value			
SDK Player Association				
SDK Version(s)	should be current SDK Version. If not, next version will be picked up on			
Prod App ID pointed to sfcode	eu-uat / eu			
based on: DCR Video - App Test Cases - v.4.5				
updated: 21.04.2020				

Test Case	Test Index	Validation	Expected Results	Pass/Fail	Value	Notes
SDK INITIALIZATION						
			apid: [correct App ID]		*provide value*	
			sfcode: [correct collection environment]		*provide value*	
1.1 SDK Configuration	1.1.1	6.0.0 SDK Initialized	**nol_sdkDebug: 'DEBUG' - Debug logging should be enabled**		*provide value*	
VIDEO-ON-DEMAND (VOD): PLAY FULL-LENGTH EPISODE						
	1.2.1	loadMetadata event is called for content	**Event: loadMetadata, metadata**			
			clientid: [assigned Nielsen client ID - if client is overriding]		*provide value (if passed)*	
			subbrand: [assigned Nielsen subbrand - if client is overriding]		*provide value (if passed)*	
			type: 'content'		*provide value*	
			program: [program name]			
		Required content metadata is passed in loadMetadata and is accurate No Special characters allowed (https://engineeringportal.nielsen.com/docs/Special_Characters)	**assetid: [unique asset ID]**		*provide value*	
			program: [program name]		*provide value*	
			title: [episode title]		*provide value*	
			length: [length of content in seconds] **Value for livestream: 86400 (if the value is unknown, please provide approximate length)**		*provide value*	
			hasAds : "0": [Non-ad-supported subscription models.2 "1": Ad-supported subscription models.] - to be tested for all available subscription models		*provide value*	
1.2 Content Metadata	1.2.2		**c-variables**		*provide value*	
NO PLAYHEADS BEFORE PREROLL (if applicable)						
1.3 Content Playheads	1.3.1	No playheads were passed between content metadata and preroll metadata (if applicable)				
PLAY PREROLL AD (if exists)						
	1.4.1	loadMetadata event is called for ad	**Event: loadMetadata, metadata**			
			assetid: [any value]		*provide value*	
	1.4.2	Required Ad Metadata is passed in loadMetadata and is acc	**type: 'preroll' or 'ad'**		*provide value*	
		playhead is passed every second for each ad	Event: playheadPosition, playhead: 1 Event: playheadPosition, playhead: 2			
	1.4.3	Note: playheadPosition should commence at 1, not 0 for brc	Event: playheadPosition, playhead: 3...[playhead for each second]			
		stop event called for each ad	**Event: stop**			
1.4 Preroll Ad	1.4.4	correct event sequence is called for ad pod or each ad	Event: loadMetadata (at ad start) Event: playheadPosition (every second for ad) Event: stop (at ad end)			
	1.5.1	loadMetadata event is called for content	**Event: loadMetadata, metadata**			
			clientid: [assigned Nielsen client ID - if client is overriding]		*provide value (if passed)*	
			subbrand: [assigned Nielsen subbrand - if client is overriding]		*provide value (if passed)*	
			type: 'content'		*provide value*	
			assetid: [unique asset ID]		*provide value*	
			program: [program name]		*provide value*	
			title: [episode title]		*provide value*	
			length: [length of content in seconds] **Value for livestream: 86400 (if the value is unknown, please provide approximate length)**		*provide value*	
			hasAds : "0": [Non-ad-supported subscription models.2 "1": Ad-supported subscription models.] - to be tested for all available subscription models		*provide value*	
1.5 Content Metadata: Second Va	1.5.2	Content metadata is **the same as original metadata** passe	**c-variables**		*provide value*	
1.6 Content Playhead	1.6	playhead is passed every second starting at 1 second *Note: playheadPosition should commence at 1, not 0 for browser integrations.*	Event: playheadPosition, playhead: 1 Event: playheadPosition, playhead: 2 Event: playheadPosition, playhead: 3...[playhead for each second]			
VIEW PING AFTER CONTENT STARTS						
1.7 View Ping	1.7.1	View ping after preroll	You should only receive a view ping after the preroll, once content starts.			
CONTINUE PLAYING CONTENT UNINTERRUPTED UNTIL DURATION PING						
			at='timer'		*provide value*	
			rt='video'		*provide value*	
			c27=cln,[total number of seconds that content played]		*provide value*	
			c51=adl, [total number of seconds that played for the ads within preroll ad pod (if ther		*provide value*	

Color Codes
testing steps
critical
initialization
content
ads
pings (checked on Nielsen's side)
optional metadata
required in certain scenarios/optional
required in dev sanity
always required (dev sanity, app update, post-prod)

Test Case	Test Index	Validation	Expected Results	Pass/Fail	Value	Notes	Color Codes
1.8 Duration Ping	1.8.1	Duration ping fires after 5 minutes	c52=noad, [number of ads played within preroll ad pod (if there were preroll ads)]		*provide value*		
SCRUB BACKWARD							
	1.9.1	provide timestamp and play content for at least 30 seconds	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
		stop event may be called in some players, is not required th	Event: stop				
	1.9.2	new playhead position is accurate (position decreases)	Event: playheadPosition, playhead: [new position]				
1.9 Scrub Backward	1.9.3	playhead continues to be passed from new position	Event: playheadPosition, playhead: [new position] Event: playheadPosition, playhead: [position increments 1 second] Event: playheadPosition, playhead: [position increments 1 second] ...				
FORCE OR WAIT FOR A DURATION PING							
			c27=cln,[total number of seconds that content played. Make sure to count time before		*provide value*		
		Duration ping fires after scrubbing	c51=adl, [total number of seconds that played for any ads after/before scrubbing (if th		*provide value*		
1.10 Duration Ping	1.10.1		c52=noad, [number of ads played after/before scrubbing (if there were ads)]		*provide value*		
SCRUB BACK TO BEGINNING OF CONTENT							
	1.11.1	provide timestamp and play content for at least 30 seconds after scrubbing	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
		stop event may be called in some players, is not required th	Event: stop				
		playhead is passed every second starting at 1 second					
1.11 Scrub Back to Beginning	1.11.2	*Note: playheadPosition should commence at 1, not 0 for browser integrations.*	Event: playheadPosition, playhead: 1 Event: playheadPosition, playhead: 2 Event: playheadPosition, playhead: 3...[playhead for each second]				
NO VIEW PING AFTER SCRUBBING TO BEGINNING							
1.12 No View Ping	1.12.1	Should not receive a new view ping after scrubbing to beginning	Pass = No view ping Fail = View ping				
SCRUB FORWARD							
	1.13.1	provide timestamp and play content for at least 30 seconds after scrubbing	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
		stop event may be called in some players, is not required th	Event: stop				
	1.13.2	new playhead position is accurate (*position increases)*	Event: playheadPosition, playhead: [new position]				
1.13 Scrub Forward	1.13.3	playhead continues to be passed from new position	Event: playheadPosition, playhead: [new position] Event: playheadPosition, playhead: [position increments 1 second] Event: playheadPosition, playhead: [position increments 1 second] ...				
FORCE OR WAIT FOR A DURATION PING							
			c27=cln,[total number of seconds that content played. Make sure to count time before and after scrub]		*provide value*		
		Duration ping fires after scrubbing	c51=adl, [total number of seconds that played for any ads after/before scrubbing (if there were ads)]		*provide value*		
1a.14 Duration Ping	1a.14.1	.	c52=noad, [number of ads played after/before scrubbing (if there were ads)]		*provide value*		
PLAY MIDROLL AD (if exists)							
	1.15.1	playhead for content stops being passed	Event: playheadPosition, playhead: [last position in seconds]				
	1.15.2	loadMetadata event is called for each ad	**Event: loadMetadata, metadata**				
			assetid: [unique Ad ID]		*provide value*		
	1.15.3	Required Ad Metadata is passed in loadMetadata and is ac	**type: 'midroll' or 'ad'**		*provide value*		
		playhead is passed every second for each ad	Event: playheadPosition, playhead: 1 Event: playheadPosition, playhead: 2				
	1.15.4	Note: playheadPosition should commence at 1, not 0 for bro	Event: playheadPosition, playhead: 3...[playhead for each second]				
	1.15.5	stop even called for each ad	**Event: stop**				
1.15 Midroll Ad Break	1.15.6	correct event sequence is called for ad pod or each ad	Event: loadMetadata (at ad start) Event: playheadPosition (every second for ad) Event: stop, (at ad end)				
	1.16.1	loadMetadata event is called for content	**Event: loadMetadata, metadata**				
			clientid: [assigned Nielsen client ID - if client is overriding]		*provide value (if passed)*		
			subbrand: [assigned Nielsen subbrand - if client is overriding]		*provide value (if passed)*		
			type: 'content'		*provide value*		
			assetid: [unique asset ID]		*provide value*		
			program: [program name]		*provide value*		
			title: [episode title]		*provide value*		

Test Case	Test Index	Validation	Expected Results	Pass/Fail	Value	Notes	Color Codes
			length: [length of content in seconds] Value for livestream: 86400 (if the value is unknown, please provide approximate length)		*provide value*		
			hasAds : "0": [Non-ad-supported subscription models.2 "1": Ad-supported subscription models.] - to be tested for all available subscription models		*provide value*		
1.16 Content Resumes following I	1.16.2	Content metadata is **the same as original metadata** passe	**c-variables**		*provide value*		
SCRUB BACKWARD ACROSS MIDROLL AD TO CHECK FOR PROPER CREDITING (*if applicable*)							
	1.17.1	provide timestamp after scrubbing	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
	1.17.2	playhead for content stops being passed	Event: playheadPosition, playhead: [last position in seconds]				
	1.17.3	loadMetadata event is called for each ad	Event: loadMetadata, metadata				
	1.17.4	Required Ad Metadata is passed in loadMetadata and is accurate	**assetid: [unique Ad ID]**		*provide value*		
			type: 'midroll' or 'ad'		*provide value*		
	1.17.5	playhead is passed every second for each ad *Note: playheadPosition should commence at 1, not 0 for browser integrations.*	Event: playheadPosition, playhead: 1 Event: playheadPosition, playhead: 2 Event: playheadPosition, playhead: 3...[playhead for each second]				
	1.17.6	stop even called for each ad	**Event: stop**				
1.17 Midroll Ad Break	1.17.7	correct event sequence is called for ad pod or each ad	Event: loadMetadata (at ad start) Event: playheadPosition (every second for ad) Event: stop, (at ad end)				
	1.18.1	loadMetadata event is called for content	**Event: loadMetadata, metadata**				
			clientid: [assigned Nielsen client ID - if client is overriding]		*provide value (if passed)*		
			subbrand: [assigned Nielsen subbrand - if client is overriding]		*provide value (if passed)*		
			type: 'content'		*provide value*		
			assetid: [unique asset ID]		*provide value*		
			program: [program name]		*provide value*		
			title: [episode title]		*provide value*		
			length: [length of content in seconds] Value for livestream: 86400 (if the value is unknown, please provide approximate length)		*provide value*		
			hasAds : "0": [Non-ad-supported subscription models.2 "1": Ad-supported subscription models.] - to be tested for all available subscription models		*provide value*		
	1.18.2	Content metadata is the **same as original metadata** passed during initial loadMetadata call	**c-variables**		*provide value*		
1.18 Content Resumes following Midroll	1.18.3	Playhead continues to be passed from resumed content position	Event: playheadPosition, playhead: [resumed content position] Event: playheadPosition, playhead: [position increments 1 second] Event: playheadPosition, playhead: [position increments 1 second] ...				
PAUSE CONTENT							
	1.19.1	provide timestamp and pause for 30 seconds	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
	1.19.2	Playhead is not passed	Event: playheadPosition, playhead: [last position of content in seconds] playhead does not continue to be sent				
1.19 Content Paused	1.19.3	Stop Call	**Event: stop**				
WAIT FOR A DURATION PING							
			c27=cln,[total number of seconds that content played. Make sure to count time before and after scrub]		*provide value*		
		Duration ping fires after resuming content	c51=adl, [total number of seconds that played for any ads after/before scrubbing (if there were ads)]		*provide value*		
1.20 Duration Ping	1.20.1		c52=noad, [number of ads played after/before scrubbing (if there		*provide value*		
RESUME CONTENT PLAYBACK							
	1.21.1	loadMetadata should be called after pause	**Event: loadMetadata, metadata**				
1.21 Content Resumed	1.21.2	playhead continues to be passed from resumed content pos	Event: playheadPosition, playhead: [resumed content position] Event: playheadPosition, playhead: [position increments 1 second] Event: playheadPosition, playhead: [position increments 1 second] ...				
SCRUB FORWARD NEAR THE END OF CONTENT AND RESUME PLAYBACK							
	1.22.1	provide timestamp and play content for at least 30 seconds after scrubbing	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		

Test Case	Test Index	Validation	Expected Results	Pass/Fail	Value	Notes	Color Codes
	1.22.2	stop event may be called in some players, is not required th	Event: stop				
	1.22.3	playhead continues to be passed from new position	Event: playheadPosition, playhead: [new position] Event: playheadPosition, playhead: [position increments 1 second] Event: playheadPosition, playhead: [position increments 1 second] ...				
	1.22.4	final playhead position is passed *Note: Make sure that the final playhead does not exceed the length of the video represented as SD in the pings.*	Event: playheadPosition, playhead: [final playhead position]				
1.22 Content Completes	1.22.5	end event called at completion of content playback	**Event: end, playhead: [final playhead position]**				
DURATION PING AT CONCLUSION OF CONTENT							
			c27=[number of seconds that played since last duration ping]		*provide value*		
			c51=adl, [total number of seconds that played for the ads within any ad pods that play		*provide value*		
1.23 Duration Ping	1.23.1	Duration ping fires at conclusion of Content	c52=noad, [number of ads played within ad pods]		*provide value*		
PLAY POSTROLL AD (if exists)							
	1.24.1	loadMetadata event is called for each ad	**Event: loadMetadata, metadata**				
			assetid: [unique Ad ID]		*provide value*		
	1.24.2	Required Ad Metadata is passed in loadMetadata and is acc	**type: 'postroll' or 'ad'**		*provide value*		
	1.24.3	playhead is passed every second for each ad Note: playheadPosition should commence at 1, not 0 for brc	Event: playheadPosition, playhead: 1 Event: playheadPosition, playhead: 2 Event: playheadPosition, playhead: 3...[playhead for each second]				
	1.24.4	stop even called for each ad	**Event: stop**				
1.24 Postroll Ad	1.24.5	correct event sequence is called for each ad	Event: loadMetadata (at ad start) Event: playheadPosition (every second for ad) Event: stop, (at ad end)				
DURATION PING AFTER POSTROLL (if applicable)							
			c27=0 [Final duration ping must be 0 for postroll]		*provide value*		
			c51=adl,[total number of seconds that played for postroll ad]		*provide value*		
1.25 Duration Ping	1.25.1	Duration ping fires at conclusion of postroll. c27 should equ	c52=noad,[number of ads played within postroll ad pod (if there were postroll ads)]		*provide value*		
RESTART VIDEO (if applicable)							
	1.26.1	loadMetadata event is called for content	**Event: loadMetadata, metadata**				
			clientid: [assigned Nielsen client ID - if client is overriding]		*provide value (if passed)*		
			subbrand: [assigned Nielsen subbrand - if client is overriding]		*provide value (if passed)*		
			type: 'content'		*provide value*		
			assetid: [unique asset ID]		*provide value*		
			program: [program name]		*provide value*		
			title: [episode title]		*provide value*		
			length: [length of content in seconds] Value for livestream: 86400 (if the value is unknown, please provide approximate length)		*provide value*		
		Required content metadata is passed in loadMetadata and	**hasAds : "0": [Non-ad-supported subscription models.2 "1": Ad-supported subscription models.] - to be tested for all available subscription models**		*provide value*		
1.26 Content Metadata	1.26.2	No Special characters allowed (https://engineeringportal.nie	**c-variables**		*provide value*		
NO PLAYHEADS BEFORE PREROLL AFTER REPLAY (if applicable)							
1.27 Content Playheads	1.27.1	No playheads were passed between content metadata and preroll metadata (if applicable)					
PLAY PREROLL AD AFTER REPLAY (if exists)							
	1.28.1	loadMetadata event is called for ad	**Event: loadMetadata, metadata**				
			type: 'preroll' or 'ad'		*provide value*		
	1.28.2	Required Ad Metadata is passed in loadMetadata and is acc	**assetid: [any value]**		*provide value*		
	1.28.3	playhead is passed every second for each ad Note: playheadPosition should commence at 1, not 0 for brc	Event: playheadPosition, playhead: 1 Event: playheadPosition, playhead: 2 Event: playheadPosition, playhead: 3...[playhead for each second]				
		stop event called for each ad	**Event: stop**				
1.28 Preroll Ad	1.28.4	correct event sequence is called for ad pod or each ad	Event: loadMetadata (at ad start) Event: playheadPosition (every second for ad) Event: stop				
	1.29.1	loadMetadata event is called for content	**Event: loadMetadata, metadata**				
			clientid: [assigned Nielsen client ID - if client is overriding]		*provide value (if passed)*		
			subbrand: [assigned Nielsen subbrand - if client is overriding]		*provide value (if passed)*		

Test Case	Test Index	Validation	Expected Results	Pass/Fail	Value	Notes	Color Codes
			type: 'content'		*provide value*		
			assetid: [unique asset ID]		*provide value*		
			program: [program name]		*provide value*		
			title: [episode title]		*provide value*		
			length: [length of content in seconds] **Value for livestream: 86400 (if the value is unknown, please provide approximate length)**		*provide value*		
			hasAds : "0": [Non-ad-supported subscription models.2 **"1": Ad-supported subscription models.] - to be tested for all available subscription models**		*provide value*		
1.29 Content Metadata: Second V	1.29.2	Content metadata is the same as original metadata passed	**c-variables**		*provide value*		
VIEW PING AFTER CONTENT REPLAY							
1.30 View Ping	1.30.1	View ping after preroll	You should only receive a view ping AFTER the preroll, once content starts.				
Interruption Scenarios							
SCENARIO #1							
EXIT PAGE DURING CONTENT PLAYBACK							
	2.1.1	provide timestamp and play content for at least 30 seconds	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
2.1 Exit page during content playback	2.1.2	playhead passed every second	Event: playheadPosition, playhead: [position increments 1 second] Event: playheadPosition, playhead: [position increments 1 second] Event: playheadPosition, playhead: [position increments 1 second] ...				
NAVIGATE TO NEW PAGE DURING CONTENT PLAYBACK							
	2.2.1	Playheads stop when leaving page	Event: playheadPosition, playhead: [last position of content in seconds] playhead does not continue to be sent				
2.2 End Event after Leaving	2.2.2	End event after leaving page	Event: end Note: The SDK should automatically pick up on this event. However, you will not always see it in the logs due the nature of the test. Likewise, you may not receive a D ping after navigating away. This is a known browser limitation with the SDK.				
DURATION PING AFTER LEAVING PAGE							
		Duration ping fires after leaving page	c27=[number of seconds that played since last duration ping]		*provide value*		
			c51=adl, [total number of seconds that played for the ads within any ad pods that played]		*provide value*		
2.3 Duration Ping	2.3.1		c52=noad, [number of ads played within ad pods]		*provide value*		
SCENARIO #2							
CLOSE WINDOW OR TAB DURING CONTENT PLAYBACK (SAVE LOGS BEFORE THIS TEST)							
	2.4.1	provide timestamp and play content for at least 30	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
	2.4.2	playheads do not continue to be sent	Event: playheadPosition, playhead: [position increments 1 second] Event: playheadPosition, playhead: [position increments 1 second] No more playheads sent after leaving page...				
2.4 Close window during content playback	2.4.3	End event after leaving page	Event: end Note: The SDK should automatically pick up on this event. However, you will not always see it in the logs due the nature of the test. Likewise, you may not receive a D ping after navigating away. This is a known browser limitation with the SDK.				
DURATION PING AFTER LEAVING PAGE							
		Duration ping fires after leaving page	c27=[number of seconds that played since last duration ping]		*provide value*		
			c51=adl, [total number of seconds that played for the ads within any ad pods that played]		*provide value*		
2.5 Duration Ping	2.5.1		c52=noad, [number of ads played within ad pods]		*provide value*		
SCENARIO #3							
PLAY NEW VIDEO							
2.6 Play New Video	2.6.1	loadMetadata event is called for content	**Event: loadMetadata**				
PLAY PREROLL AD (*if exists*)							
2.7 Play preroll ad	2.7.1	correct event sequence is called for ad pod or each ad	Event: loadMetadata Event: playheadPosition, playhead: 0...[playhead for each second]				
LEAVE PAGE AS PREROLL AD PLAYS							
2.8 Exit page during preroll	2.8.1	provide timestamp and play content for at least 5 seconds	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		

Test Case	Test Index	Validation	Expected Results	Pass/Fail	Value	Notes
NO VIEW PING AFTER LEAVING PAGE						
2.9 No View Ping	2.9.1	Should not receive a new view ping after leaving during preroll	Pass = No view ping Fail = View ping			
DURATION PING AFTER LEAVING PAGE						
2.10 Duration Ping	2.10.1	Duration ping fires after leaving during preroll	c27=0		*provide value*	
			c51=adl, [total number of seconds that played for the ads within any ad pods that played]		*provide value*	
			c52=noad, [number of ads played within ad pods]		*provide value*	
SCENARIO #4						
ENABLE NETWORK THROTTLING DURING PREROLL						
2.11 Content Metadata	2.11.1	loadMetadata event is called for content	**Event: loadMetadata, metadata**			
NO PLAYHEADS BEFORE PREROLL (*if applicable*)						
2.12 Content Playheads	2.12.1	No playheads were passed between content metadata and preroll metadata (if applicable)				
PLAY PREROLL AD (*if exists*)						
2.13 Preroll Ad	2.13.1	loadMetadata event is called for ad	**Event: loadMetadata, metadata**			
	2.13.2	playhead is passed every second for each ad	Event: playheadPosition, playhead: 1 Event: playheadPosition, playhead: 2 Event: playheadPosition, playhead: 3...[playhead for each second]			
2.14 Network Throttling	2.14.1	provide timestamp at the point where buffering has completed and the player has stopped	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*	
	2.14.2	Playheads stop after buffered video	Event: playheadPosition, playhead: [increments every 1 second starting at 0] *Playhead does not continue to be sent*			
	2.14.3	stop event should be sent	**Event: stop**			
DISABLE NETWORK THROTTLING						
2.15 Network Speed Restoration	2.15.1	provide timestamp at the point where the player resumes. **No Loadmetadata Content should be fired**	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*	
	2.15.2	playheads resume after network restoration	Event: playheadPosition, playhead: [resumes from last position] Event: playheadPosition, playhead: [increments every 1 second]			
2.16 Preroll Ad	2.16.3	stop event called for each ad	**Event: stop**			
	2.16.4	correct event sequence is called for ad pod or each ad	Event: loadMetadata (at ad start) Event: playheadPosition (every second for ad) Event: stop (at ad end)			
2.17 Content Metadata: Second Validation	2.17.1	loadMetadata event is called for content	**Event: loadMetadata, metadata**			
2.18 Content Playhead	2.18.1	playhead is passed every second starting at 1 seconds	Event: playheadPosition, playhead: 1 Event: playheadPosition, playhead: 2 Event: playheadPosition, playhead: 3...[playhead for each second]			
VIEW PING AFTER CONTENT STARTS						
2.19 View Ping	2.19.1	View ping after preroll (if applicable)	You should only receive a view ping after the preroll, once content starts.			
SCENARIO #5						
ENABLE NETWORK THROTTLING DURING CONTENT						
2.20 Network Throttling	2.20.1	provide timestamp at the point where buffering has completed and the player has stopped	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*	
	2.20.2	Playheads stop after buffered video	Event: playheadPosition, playhead: [increments every 1 second] *Playhead does not continue to be sent*			
	2.20.3	stop event should be sent	**Event: stop**			
DISABLE NETWORK THROTTLING						
2.21 Network Speed Restoration	2.21.1	loadMetadata event is called for content	**Event: loadMetadata, metadata**			
	2.21.2	provide timestamp at the point where the player resumes	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*	
	2.21.3	playheads resume after network restoration	Event: playheadPosition, playhead: [resumes from last position] Event: playheadPosition, playhead: [increments every 1 second]			
SCENARIO #6						
USE SETTINGS FEATURES IF APPLICABLE (SUBTITLES, VIDEO QUALITY CHANGE, CHANGE AUDIO SETTINGS, ETC)						

Color Codes

Test Case	Test Index	Validation	Expected Results	Pass/Fail	Value	Notes	Color Codes
2.22 Change settings during content playback (add subtitles, change video quality, etc.)	2.22.1	provide timestamp at the point where buffering has completed and the player has stopped	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
	2.22.2	Playheads stop after buffered video	Event: playheadPosition, playhead: [increments every 1 second] *Playhead does not continue to be sent*				
	2.22.3	stop event should be sent	**Event: stop**				
SCENARIO #7							
OPEN VIDEO IN FULLSCREEN							
2.23 open fullscreen	2.23.1	provide timestamp at the point where buffering has completed and the player has stopped	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
	2.23.2	Playheads stop after buffered video	Event: playheadPosition, playhead: [increments every 1 second] *Playhead does not continue to be sent*				
	2.23.3	stop event should be sent if video pauses playheads should be passed if video continues	**Event: stop** **Playheads should be passed if video continues**				
SCENARIO #8							
GO TO PICTURE IN PICTURE MODE							
2.24 open fullscreen	2.24.1	provide timestamp at the point where buffering has completed and the player has stopped	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
	2.24.2	Playheads stop after buffered video	Event: playheadPosition, playhead: [increments every 1 second] *Playhead does not continue to be sent*				
	2.24.3	stop event should be sent if video pauses playheads should be passed if video continues	**Event: stop** **Playheads should be passed if video continues**				
SCENARIO #9							
CLOSE PIP PER X							
2.25 close PiP per X	2.25.1	provide timestamp at the point where buffering has completed and the player has stopped	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
	2.25.2	Playheads stop after buffered video	Event: playheadPosition, playhead: [increments every 1 second] *Playhead does not continue to be sent*				
	2.25.3	end should be called if video closes stop should be called if videos pauses	Event: end Note: call end if video is closed after hitting X. Event: stop Note: call stop if video continues				
SCENARIO #10							
REFRESH PAGE DURING PLAYBACK							
2.26 refresh page during playback	2.26.1	provide timestamp and play content for at least 30	Copy timestamp or paste the line into column H. In the Console tab of your browser, run this command to get a timestamp: **Date('now')**.		*provide value*		
	2.26.2	playheads do not continue to be sent	Event: playheadPosition, playhead: [position increments 1 second] Event: playheadPosition, playhead: [position increments 1 second] No more playheads sent after leaving page...				
	2.26.3	End event after leaving page	Event: end Note: The SDK should automatically pick up on this event. However, you will not always see it in the logs due the nature of the test. Likewise, you may not receive a D ping after navigating away. This is a known browser limitation with the SDK.				
2.26 Duration Ping	2.26.3	Duration ping fires after leaving page	c27=[number of seconds that played since last duration ping]		*provide value*		
			c51=adl, [total number of seconds that played for the ads within any ad pods that played]		*provide value*		
			c52=noad, [number of ads played within ad pods]		*provide value*		