WebSDK error handling

When 'FAILED' session, stream or call status is received, Stream.getInfo() method returns a description string of error occured.

Errors list

Error	Description string
streamStatusInfo	
FAILED_BY_ICE_ERROR	Failed by ICE error
FAILED_BY_ICE_TIMEOUT	Failed by ICE timeout
FAILED_BY_KEEP_ALIVE	Failed by ICE keep alive
FAILED_BY_DTLS_FINGERPRINT_ERROR	Failed by DTLS fingerprint error
FAILED_BY_DTLS_ERROR	Failed by DTLS error
FAILED_BY_HLS_WRITER_ERROR	Failed by HLS writer error
FAILED_BY_RTMP_WRITER_ERROR	Failed by RTMP writer error
FAILED_BY_RTP_ACTIVITY	Failed by RTP activity
STOPPED_BY_SESSION_DISCONNECT	Stopped by session disconnect
STOPPED_BY_REST_TERMINATE	Stopped by rest /terminate
STOPPED_BY_PUBLISHER_STOP	Stopped by publisher stop
STOPPED_BY_USER	Stopped by user
FAILED_BY_ERROR	Failed by error
FAILED_TO_ADD_STREAM_TO_PROXY	Failed to add stream to proxy
DISTRIBUTOR_STOPPED	Distributor stopped
PUBLISH_STREAM_IS_NOT_READY	Publish stream is not ready
STREAM_NOT_FOUND	Stream not found
STREAM_NAME_ALREADY_IN_USE	Stream name is already in use
MEDIASESSION_ID_NULL	MediaSessionId is null
MEDIASESSION_ID_ALREADY_IN_USE	MediaSessionId is already in use
SESSION_NOT_READY	Session not ready
SESSION_DOES_NOT_EXIST	Session does not exist
RTSP_HAS_WRONG_FORMAT	Rtsp has wrong format
FILE_HAS_WRONG_FORMAT	File has wrong format
FAILED_TO_CONNECT_TO_RTSP_STREAM	Failed to connect to rtsp stream
RTSP_STREAM_NOT_FOUND	Rtsp stream not found
RTSPAGENT_SHUTDOWN	RtspAgent shutdown
STREAM_FAILED	Stream failed
NO_COMMON_CODECS	No common codecs
BAD_URI	Bad URI
GOT_EXCEPTION_WHILE_STREAMING_FILE	Got exception while streaming file
REQUESTED_STREAM_SHUTDOWN	Requested stream shutdown

FAILED_TO_READ_FILE	Failed to read file
FILE_NOT_FOUND	File not found
FAILED_TO_CONNECT_TO_ORIGIN_STREAM	Failed to connect to origin stream
CDN_STREAM_NOT_FOUND	CDN stream not found
FAILED_TO_GET_AGENT_STORAGE	Failed to get agent storage
AGENT_SERVICING_ORIGIN_STREAM_IS_SHUTTING_DOWN	Agent servicing origin stream is shutting down
TERMINATED_BY_KEEP_ALIVE	Terminated by keep-alive
TRANSCODING_REQUIRED_BUT_DISABLED	Transcoding is requred, but disabled
NO_AVAILABLE_TRANSCODERS	No available transcoder nodes in CDN
callStatusInfo	
FAILED_BY_SESSION_CREATION	Failed by session creation
FAILED_BY_ICE_ERROR	Failed by ICE error
FAILED_BY_RTP_ACTIVITY	Failed by RTP activity
FAILED_BY_RTMP_WRITER_ERROR	Failed by RTMP writer error
FAILED_BY_DTLS_FINGERPRINT_ERROR	Failed by DTLS fingerprint error
FAILED_BY_DTLS_ERROR	Failed by DTLS error
FAILED_BY_ERROR	Failed by error
FAILED_BY_REQUEST_TIMEOUT	Failed by request timeout
TRANSCODING_REQUIRED_BUT_DISABLED	Transcoding is requred, but disabled
errorInfo	
NONE_OF_MEDIAPROVIDERS_AVAILABLE	None of MediaProviders available
NONE_OF_PREFERRED_MEDIAPROVIDERS_AVAILABLE	None of preferred MediaProviders available
FLASHPHONER_API_NOT_INITIALIZED	Flashphoner API is not initialized
OPTIONS_URLSERVER_MUST_BE_PROVIDED	options.urlServer must be provided
INVALID_SESSION_STATE	Invalid session state
OPTIONS_MUST_BE_PROVIDED	options must be provided
INVALID_CALL_STATE	Invalid call state
EVENT_CANT_BE_NULL	Event can't be null
CALLBACK_NEEDS_TO_BE_A_VALID_FUNCTION	Callback needs to be a valid function
INVALID_SESSION_STATE	Invalid session state
OPTIONS_NAME_MUST_BE_PROVIDED	options.name must be provided
CAN_NOT_SWITCH_CAM	Number of cams is less than 2 or already used custom stream
CAN_NOT_SWITCH_MIC	Number of mics is less than 2 or already used custom stream
LOCAL_ERROR	Local erroraised in browser

Local browser errors catching

When local browser error is catched, Stream.getInfo() method returns Flashphoner.constants.ERROR_INFO.LOCAL_ERROR. In this case Stream. getErrorInfo() method returns loacl error description based on exception information, for example:

- Requested device not found - all microphones or cameras are disabled
- Could not start video source - camera is busy by another application (Chromium-browser message)
- Failed to allocate videosource -camera is busy by another application (Firefox message)
- Permission denied - access to camera/mic is not allowed
- Invalid constraint - unsupported resolution is selected in Safari
- This provider doesn't support getMediaAccess - WSPlayer is used, or pablisher page is opened via insecure connection (HTTP)

Error handling code example

As an example. let's take the Two Way Streaming application code version with hash 501f72f, that is available to download in build0.5.28.2753.143

code

```
function setStatus(selector, status, stream) {
    var statusField = $(selector);
    statusField.text(status).removeClass();
    if (status == "PLAYING" || status == "ESTABLISHED" || status == "PUBLISHING") {
        ...
    } else if (status == "FAILED") {
        if (stream) {
            if (stream.published()) {
                switch(stream.getInfo()){
                    case STREAM_STATUS_INFO.STREAM_NAME_ALREADY_IN_USE:
                        $("#publishInfo").text("Server already has a publish stream with the same name, try 
using different one").attr("class", "text-muted");
                        break;
                    case ERROR_INFO.LOCAL_ERROR:
                        $("#publishInfo").text("Browser error detected: " + stream.getErrorInfo()).attr
("class", "text-muted");
                        break;
                    default:
                        $("#publishInfo").text("Other: "+stream.getInfo()).attr("class", "text-muted");
                        break;
                }
            } else {
                switch(stream.getInfo()){
                    case STREAM_STATUS_INFO.SESSION_DOES_NOT_EXIST:
                        $("#playInfo").text("Actual session does not exist").attr("class", "text-muted");
                        break;
                    case STREAM_STATUS_INFO.STOPPED_BY_PUBLISHER_STOP:
                        $("#playInfo").text("Related publisher stopped its stream or lost connection").attr
("class", "text-muted");
                        break;
                    case STREAM_STATUS_INFO.SESSION_NOT_READY:
                        $("#playInfo").text("Session is not initialized or terminated on play ordinary stream").
attr("class", "text-muted");
                        break;
                    case STREAM_STATUS_INFO.RTSP_STREAM_NOT_FOUND:
                        $("#playInfo").text("Rtsp stream not found where agent received '404-Not Found'").attr
("class", "text-muted");
                        break;
                    case STREAM_STATUS_INFO.FAILED_TO_CONNECT_TO_RTSP_STREAM:
                        $("#playInfo").text("Failed to connect to rtsp stream").attr("class", "text-muted");
                        break;
                    case STREAM_STATUS_INFO.FILE_NOT_FOUND:
                        $("#playInfo").text("File does not exist, check filename").attr("class", "text-muted");
                        break;
                    case STREAM_STATUS_INFO.FILE_HAS_WRONG_FORMAT:
                        $("#playInfo").text("File has wrong format on play vod, this format is not supported").
attr("class", "text-muted");
                        break;
                    default:
                        $("#playInfo").text("Other: "+stream.getInfo()).attr("class", "text-muted");
                        break;
                }
            }
        }
        statusField.attr("class", "text-danger");
    }
}
```