Video Knowledge Capture (VKC)
Harris Corporation

CS307, Software Engineering (Spring 2009)
Video-conferences on Wednesdays

Purpose:
To create a tool which will allows users document "knowledge" using video and audio. This tool will allow for capture, storage, and retrieval of the knowledge in video and audio format.

Summary:
This tool will record desktop activity upon user request. The captured video will be associated with user provided metadata, and stored in a central repository where the videos are searchable and accessible by their attached metadata.

Detail:
The user will install a capture utility on their personal computer. The user will be able to start recording a session, during which time everything the user does on their screen will be captured, along with microphone input if desired. At the end of a capture session, the user will be prompted for metadata to be associated with the recording. The fields for the metadata will be customizable and could include such information as: Application, Version, Description, Author, Permissions etc.

After the user has input the required metadata, the video and associated metadata will be uploaded to a central video repository server. The server will parse the metadata and organize the video accordingly. Users will be able to search for knowledge documented in video using a web interface, with searches being done on the metadata for potential matches. Users must authenticate themselves prior to searching for and viewing videos. Search results will only return videos which the current user has the authority to view. Once the user selects a video, a user will be able to download the video file or play it directly from the server. Individuals with administrative privileges will be able to modify a stored video's metadata and permissions, and remove videos from the central repository. A stored video's owner will also be able change the metadata and permissions, and delete it from the server.

Example:
A potential usage scenario would be a user starting the application and then recording a session in which they set up a new C++ project in Rational Rose. The user would stop the recording and then enter in information for the metadata tag such as Application = Rational Rose, Version = 2003, Description = Creating a new C++ project, Permission=All. After the user saved the metadata, the application would upload both the video capture (audio included) and the metadata to the central video repository server.

Another user who is looking for help on that particular topic would go to the central video repository server web portal and search for Rational Rose or creating new project. This video would show up in a list of possible matches. They would select the video,

which would link to a file share with the option of either opening the file or playing it directly from the share.

Stretch Goals:

1. Add metrics to track user uploads and downloads. High value contributors could be determined by how frequently their videos are downloaded. Videos downloaded more frequently could be placed at the top of the search listings.
2. Add the ability to stream video rather than user downloading it.
3. Videos could be tagged and encrypted.
4. Ability to add audio after the video portion has been recorded.
5. Make it platform independent

See:

-Screencast (definition) http://en.wikipedia.org/wiki/Screencast
-Java Media Framework API (JMF)
http://java.sun.com/javase/technologies/desktop/media/jmf/