CPSC 301 Lab 2 Assignment: Requirements Gathering

Jeff Wintersinger
2012.01.25

Abstract

Our client is an emergency room physician who hopes to deploy an intra-ER communications system that will allow administrators to rapidly announce new policies to staff; users to communicate using both immediate, relatively transient means such as instant messaging, as well as more persistent, topic-specific channels such as forums; staff to update a department-wide calendar with upcoming educational opportunities for residents; and an educational database with the ability for users to post treatment information, upload academic papers, and discuss content with others. A hodgepodge of poorly designed systems currently satisfy these needs, with the client intending to replace them with a single unified application that integrates with the hospital's existing keycard authentication mechanism. Users will consist of ER staff; as the application will likely be deployed city-wide, each deployed instance of the application should be specific to a given emergency room. Users, however, must have the ability to easily transition between these instances, as a given doctor may work in multiple ERs. As the only users will be trusted hospital employees, the user may be safely assumed to be non-malicious, with a minimum of segmentation in access rights required. The only two user classes required by the client are a global admin class, with the ability to add users, forums, calendar events, and otherwise manipulate the system, and a user class, with the ability to post content in almost any part of the system but not edit its structure.

As the system requires both large-scale content entry suited to full-scale computer terminals, as well as constant access by mobile users moving around the hospital environment, it should be developed as both a web application intended for devices with large screens (optimized for content entry) and as a smartphone application (optimized for mobile connectivity). As users may possess a smattering of different smartphone models, we may develop native applications for each of the most popular mobile OSes, or we may develop a mobile-optimized web application that, while offering a slightly less pleasant experience, is device agnostic. Both options should be presented to the client, with her preference likely dependent on her budget.

Functional Requirements

1. Users must have access to a calendar used to list upcoming educational opportunities, primarily for residents. The client specified that she needs the ability to add and modify scheduled events; however, as she also stated that all users are trusted not to be malicious, this capability could be extended to some or all users.
2. Users must be able to communicate in real time with one another via an instant messaging-like system. This ability should be optimized for mobile smartphone use, though the ability to access the same capabilities via computer terminals should also be provided. In addition to communication between two individuals, users must easily be able to converse within small groups, or even as a single large group including all users at the work site. Ultimately, the user's goal is to replace pagers currently used as a means of quickly reaching users
3. Users must have access to forums in which they can conduct threaded discussions on topics of

interest. Administrators must be able to easily add and remove forums as they please. Ideally, support for rich media such as video and images should be provided.
4. System administrators must have the ability to push out announcements to all users at the local work site. A suggested use case is announcing a spate of new policies for treating patient afflictions related to use of new ecstasy varieties.
5. Users must be able to post and read educational materials related to treatment. Ideally, this would take the form of a wiki, with users able to add and modify any information in the system. Users must also have the ability to upload files such as academic papers.
6. The system must permit real-time sharing of video, with a typical use case involving ultrasound video taken of patients. This video must be stored securely within the hospital's system, without relying on external video sharing services.
7. Access to the system must be possible both inside and outside the hospital. Within the hospital, smartphone users should have access to the system via wifi, given the difficulty of accessing mobile telecommunications networks from within large concrete buildings.
8. Users should be able to log in to the system at computer terminals via the established keycard system or with a traditional user name and password. (Thus, the system's user database must integrate smoothly with the external user database that drives the keycard system.) In addition to providing an explicit ability to log out, the system should automatically terminate idle sessions after a fixed period of time. If the user logged in via her keycard, her session should automatically terminate when the keycard is removed from the terminal.

Nonfunctional Requirements
1. Availability requirements are rather relaxed. The client indicated that, say, a weekly downtime of thirty minutes on a predetermined day at midnight for performing system maintenance would be perfectly acceptable.
2. No hard timeline was provided by the client, with her only provision stating that sooner was better. The client indicated a willingness for the system to be developed iteratively, with us deploying a limited solution at first that would be progressively expanded over time. In such a case, she stated that the instant messaging capability would be of secondary importance relative to the calendar, forums, announcements, and education functions. The smartphone application could thus be delayed until after the initial system is deployed as a large-screen web app.
3. Budget-wise, the client again gave no hard cap, stating only that her department could essentially request whatever funds they desired. I take this as free license to gouge the hospital —we shall at last have the chance to suckle from the sweet teat that is government spending. (Worryingly, however, the client stated that she thought $10,000 was an unreasonably high cost for an application such as this, a figure that strikes me as low even if we don't plan on abusing the public trust.)

Use Case

Summary: The user must be able to upload videos to share with other users, such as in the case of patient radiology videos.
Actor: Any user of the system.
Stakeholders: The patient and hospital administration.
Precondition: The user has a video file she wishes to share with other users. The user has an account on the system.
Postcondition: The user's video has been added to the application such that it can be viewed by other users.

Main success scenario:
1. The user (U) logs into the system with her keycard, or via her username and password. Assume that U is using a local computer terminal, though the steps would be largely the same if she were using a smartphone, albeit with interface controls specialized for the device.
2. The application's main screen is rendered. U selects the "Videos" section from the navigation.
3. The videos section appears. From here, U clicks the "Add video" link.
4. The video addition interface is rendered, allowing U to select a video file, as well as enter a title, category, and description for the video.
5. U selects a valid video file on her local computer, and also enters a valid title, description, and category. She clicks the "Upload video" button.
6. The video is transmitted to the server. A progress bar is used during the transfer to indicate the status of the upload.
7. The user is notified that her upload was successful, and that the video is being processed. The user may now log out (Step 10), or wait for the video to process.
8. The server encodes the video to a Web-friendly format so that it may be delivered via Flash or HTML5.
9. If the user remains logged into the application, the server notifies the user that processing is complete, and the video is available for viewing.
10. The user logs out of the system. If she logged in via a keycard, she is automatically logged out when she removes the card. If she logged in manually and neglects to click the "log out" link, she is automatically logged out after a predefined period of idleness.

Extension A: At Step 5, the user selected a non-video file, or entered a blank title, description, or category. The user is notified of her error, and prompted to re-enter the invalid input. Once she does, the main success scenario resumes at Step 6.

Extension B: At Step 8, an error occurs during video encoding. The server deletes the uploaded file, notifies the user of the error via the system's messaging capability, and displays a notification if the user has remained logged in while she waits for her video to process. The main success scenario terminates unsuccessfully at this step, though the user may being anew from Step 1.