

# PACS on Mobile Devices

Ashesh Parikh, Ph.D<sup>a</sup> and Nihal Mehta, Ph.D.<sup>b</sup>

<sup>a</sup>netDICOM, 11105 Latimer Drive, Frisco, TX USA;

<sup>b</sup>netDICOM, 11105 Latimer Drive, Frisco, TX USA

## ABSTRACT

Recent advances in internet browser technologies makes it possible to incorporate advanced functionality of a traditional PACS for viewing DICOM medical images on standard web browsers without the need to pre-install any plug-ins, apps or software. We demonstrate some of the capabilities of standard web browsers setting the stage for a cloud-based PACS.

**Keywords:** Mobile devices, PACS, zero footprint viewer, Cloud Computing

## 1. INTRODUCTION

The growth and power of mobile devices together with speedier networks has resulted in the biggest shift ever in technology and enhanced the human lifestyle and experience. Mobile applications are more creative in design and offer increasing utilities and functionalities that were unheard of a few years ago. Large players like Apple, Google, Qualcomm and others are fueling this revolution by releasing their leading products in the mobile space; enormous amounts of e-commerce transactions are conducted daily over secure networks without the explicit use of LANs or VPNs. Because of the proprietary nature of medical images and other clinical data, the healthcare industry has not fully embraced the power of the internet.

Recent advances in internet technologies<sup>1</sup> and hardware offer opportunities for healthcare professionals to synthesize information from a variety of sources to enhance patient treatment and improve clinical outcomes. Here, a cloud-based system that can handle medical images (DICOM's/NIFTI/JPEG's etc) using a standard web-browser on a user's device such as Apple's iPad<sup>TM</sup>, desktop computers and smartphones is described. Specifically, the DICOM medical image is rendered on any device that has a standard web-browser such as Chrome, Safari, Firefox etc. without an explicit download of any software, app or plug-in.

## 2. BROWSER TECHNOLOGIES

DICOM medical image files are encoded in a format specific to the healthcare industry and contains clinical information in addition to the pixel intensities. Clinical information includes the patient's name, address, date of birth, gender, etc. as well as parameters describing how the image was taken such as radiation dosage, etc. Consequently, web browsers found on mobile and desktop devices cannot display these image files. However, recently introduced HTML5 standard<sup>1</sup> provides the mechanisms to view DICOM's on web browsers without the need to first convert to lossy formats of standard digital images used in consumer photography such as JPG or PNG. Specifically, HTML5's <canvas> tag provides the ability to read pixel intensities specified in the bytes of a DICOM file and display on the browser.<sup>1</sup> The related specification for cascading style sheets (CSS3) allows the browser page to be configured to display multiple images as used in hanging protocols.

The DICOM viewer described here resides within a standard web-browser such as Chrome, Safari, Firefox etc., on the user's device including smartphones and tablets. The viewer has no explicit requirement to download any other software, app or plug-in to enable it for viewing.

---

Further author information: (Send correspondence to Ashesh Parikh.)

Ashesh Parikh: E-mail: ashesh.parikh@netdicom.net, Telephone: (972) 905-NET-D (6383)

Nihal Mehta: E-mail: nihal.mehta@netdicom.net, Telephone: (972) 905-NET-D (6383)

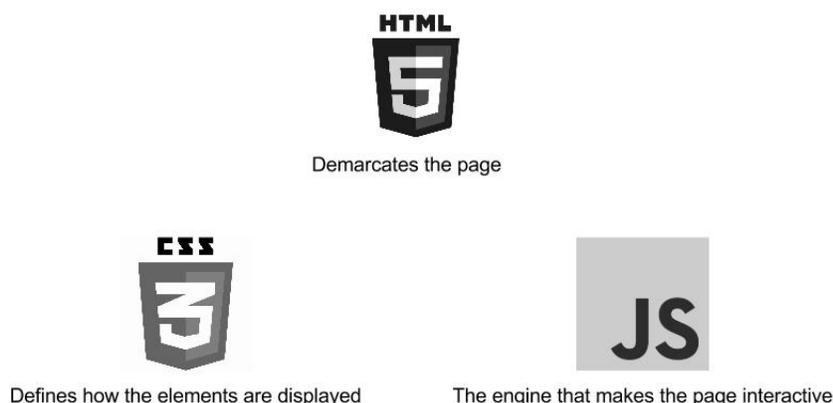


Figure 1. 3 Pillars of Web-Based Applications (Client-Side Technology)

It's not enough to simply view a single DICOM image frame on a mobile device. For a mobile device to be used in clinical practice, the mobile device must be capable of displaying the frames of a series in succession so that the user can "scroll" through them without the continual refreshing of the web page. (A page refresh, from a user's perspective, manifests as a blank page before the new page loads in the browser.)

The functionality that facilitates the scrolling through the frames in a seamless manner is Javascript's AJAX. Ajax (Asynchronous JavaScript And XML) is a group of interrelated Web development techniques used on the client-side to create asynchronous Web applications. In HTML5, AJAX has been upgraded so that binary files can be downloaded from the server without page refreshes.

These technologies are the prime drivers for making the viewer a truly zero footprint viewer. Furthermore, as this is accomplished on a web-browser, it offers many possibilities to replicate and customize the display of a traditional PACS system.<sup>2</sup> Moreover, while Radiologists may use their workstations for diagnosis and report writing in their daily work, easy access of images on any device over a secure connection anywhere can enable them to respond to emergency situations.<sup>3</sup> As a practical matter, a clinical practice instead of keeping a specialist on call, could find it more efficient to reach out to them as and when needed during emergencies. Finally, the security and permissions for users can be managed by appropriate authentication and authorization mechanisms; after the session is over, all image and other data is deleted automatically. Thus, this solution is safe and secure and nothing can be copied or transferred to another system at a later time.

The 3 pillars, see Figure 1, that enable an interactive web page are HTML, CSS and JavaScript. HTML script allows the marking of the page such as the demarcation between the text and the image area. CSS defines how the elements are displayed on the same page. Finally, JavaScript is the engine that makes the page interactive.

To support the viewing of DICOM medical images on a mobile device's web browser without the user having to download any plug-ins, apps or pre-install software or hardware graphics drivers, the DICOMs first have to be uploaded to web-based PACS application on a web server.<sup>3</sup> Such applications typically read the DICOM headers<sup>4</sup> when the DICOM files are uploaded to the system. Then they create patient records in a relational database that associate the patient and the DICOM files.

Subsequently, users are able to view worklists or search for the DICOMs of patients securely over the internet.<sup>3</sup> Once the user selects a specific series to view, the server side code on the web-based application will send an HTTP response to the requesting web browser. This response will contain the appropriate HTML/Javascript/CSS code. The HTML will contain the markup information for the DICOM, and the CSS will contain the layout information. The key, however, is the JavaScript code that contains the instructions to download the relevant DICOM medical image files, and then to extract the pixel intensities and display them within the HTML5 `<canvas>` tag.

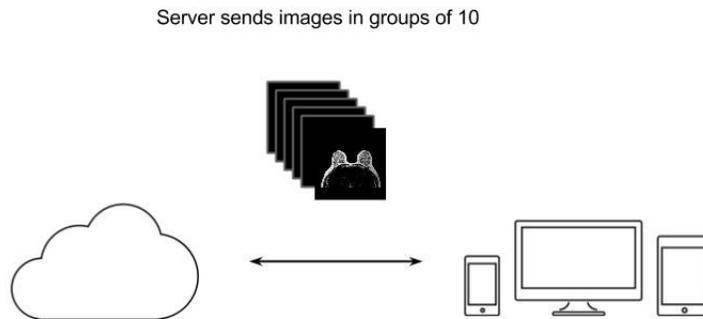


Figure 2. Sending DICOM Image Files to Web Browsers

Figure 2 is a schematic of DICOM image files that are being sent to the web browser from a web-based PACS application.

Nowadays, most users around the world enjoy high broadband upload and download rates (in Mbps). As a result, on a typical wi-fi network, an entire series of CT study comprising of about 130 images can be downloaded in about 30 seconds and displayed on a web browser. Consequently, DICOMs images can be rendered in almost real-time. However, to further decrease the latency and have the images display even faster, several optimization techniques can be used. For example, web-based applications can be configured to download the image files for the first few frames first and then downloading the others as the user starts to scroll through the frames. Rendering can be sped up even more. Instead of sending all the bytes of the DICOM image file, only a fraction of the pixel intensities can be sent to the web browser initially. This reduces the payload that needs to be sent initially. This process will result in a lower resolution image. However, there would be no visible degradation since the missing pixel values are needed only for zooming. Then, once the image has been rendered, the other bytes are sent from the web-based PACS application to the web browser.

Likewise, upload rates have also increased over the last few years. For example, it's fairly common to upload 30 CT images per minute to the web-based PACS. Using compression, even higher upload speeds can be achieved.

### 3. WEB-BASED DICOM VIEWER

In this section, we illustrate some of the capabilities of an advanced DICOM viewer in a standard web browser, rivaling those of a traditional PACS workstation. In all of the examples shown, the diagnostic quality DICOM medical images are displayed without the user having to download and install any plug-ins, apps or any other software or graphics drivers.

#### 3.1 Examples of Studies Displayed on Viewer

Figure 3 shows 2 series from a single study of an MRI Breast Bilateral WWO; one is a CAD 103 sub phase 4 and other is a dADC. The Radiologist can scroll over the series by using 2 fingers of the touch pad or by using the mouse-wheel. Full functionality is available for image viewing; access to standard viewing tools is also available, see the toolbar at the top of each image. Using these tools, the user can adjust the image WW/WC, zoom level, pan across the image, as well as measure various geometrical quantities such as length, shape, and angle.

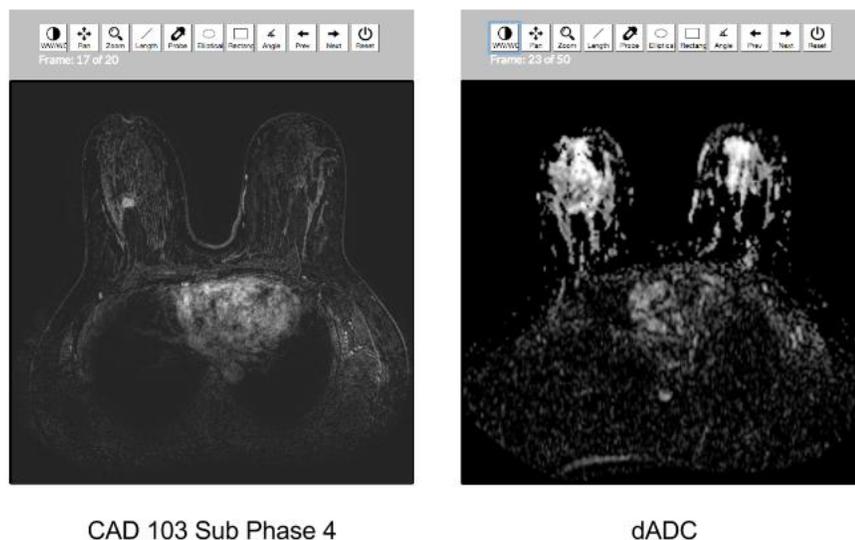


Figure 3. Display of 2 Series from a Single Study (Actual screenshot taken from netDICOM.net's netVIEWER, a commercial viewer)

Figure 4 shows an example where 4 series are displayed simultaneously on a web-based Viewer. Upper-left shows a recent series of MRI Bilateral with T without Contrast. Lower-left is a PET CT Tumor Whole Body, while the right-side has been split to display 2 series from the base-line MRI Bilateral T without Contrast. This illustrates the flexibility that is offered by using a web-based architecture.

### 3.2 Volumetric Reconstruction

Since the DICOM image is displayed on the web browser using HTML5's `<canvas>` tag, the pixel intensities can be manipulated using JavaScript. Consequently, advanced image processing algorithms can be applied to highlight clinical regions of interest. Specifically, convolution kernels enable image transformations to detect edges of tumors and other medically significant artifacts. Other algorithms such as Maximum Intensity Projection, Multi-Planar Reconstruction, and Edge Detection can also be applied for additional analysis. Figure 5 demonstrates that the browser is capable of running edge detection algorithm. The image on the left is the original DICOM image. The image on the right is the result of executing a basic edge detection algorithm within the browser itself without any plug-ins or pre-installed software. (In practice, as in this example too, a series of image processing algorithms are executed in sequence. Typically, smoothing kernels are used first, often followed by thresholding to remove unwanted artifacts in the image before executing the edge detecting algorithms themselves. In addition, the kernel specifications differ when detecting tumors and lesions in different organs or tissues. All of these variances can be accommodated in a web-based PACs.)

It's important to note that the native image resolution of browsers, and many other PACS workstations, is limited to  $2^8 = 256$  shades of gray. So, while image processing kernels can be applied using JavaScript on the DICOM image in the browser itself, it may not be sufficient for certain clinical uses. For example, a typical CT scan produces DICOM images of higher resolutions  $2^{16} = 65,536$  shades of gray.<sup>4</sup> In these cases, the image processing algorithms to extract relevant clinical information can be run on the server. The server-side code would apply the appropriate kernels against the pixel intensities extracted from the DICOM itself. Thus, medically significant information, often invisible to the naked eye, can be uncovered. It's worthwhile to point

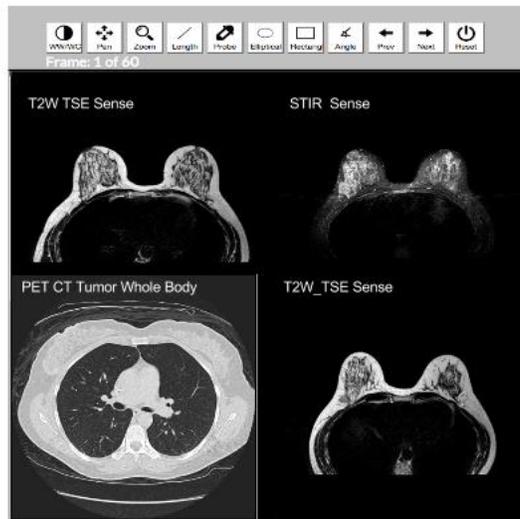


Figure 4. Multiple Series (Actual screenshot of a commercial viewer, netDICOM's netVIEWER )

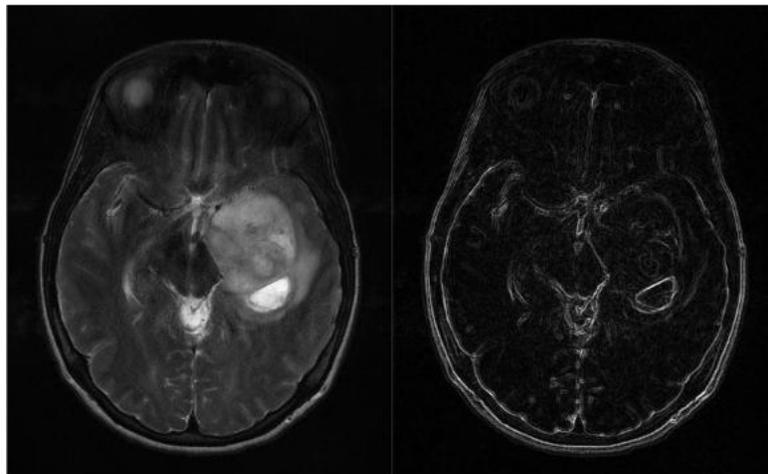


Figure 5. Detecting Tumors using Edge Detection (Actual screenshot taken from netDICOM.net's netVIEWER, a commercial viewer)

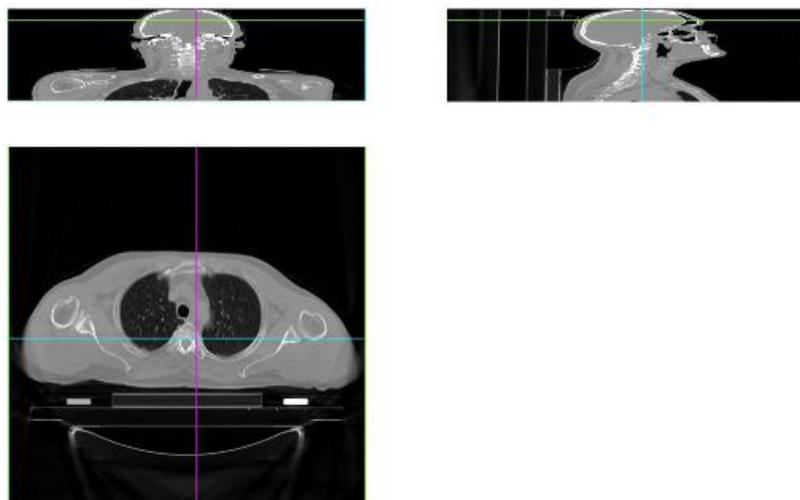


Figure 6. Multiplane Reconstruction (Actual screenshot taken from netDICOM.net's netVIEWER, a commercial viewer)

out that most radiologists work within the limitations of 256 shades of gray. In developing countries, where PACS workstations are few, the DICOMs medical images are first converted to lossy JPG images. Thus, using current web-based technologies, not only can these countries benefit from viewing DICOMs in their native form, physicians all over the world can “see” information that was previously not visible to them.

In addition, kernels that simply enhance the user experience by using image data for planar reconstruction can still be performed on the browser. Figure 6 shows an example of Multi-Planar reconstruction of an image.

Modalities where image slices with very small thickness are collected in one axis can be used to reconstruct image view's in other axes. In the figure, CT's from rlog conversion with Axial view for Head-Neck region were collected. Using the Multi-Planar algorithm, the two other planes viz. Coronal and Sagittal have been derived and presented on the viewer. Image Processing enables transformation and reconstruction of data from tools with high spatial resolution and project the data in other views for better diagnosis.

### 3.3 Hanging Protocols

A useful feature of traditional PACS workstations is hanging protocols. These controls the arrangement of different viewing areas so that the radiologist can see different aspects of the study simultaneously while reading and reporting.

Hanging protocols can be replicated easily on a web page using CSS, including the ability to flip images, especially useful for breast studies.

Figure 7 is a schematic of how a user can define several windows. Using CSS, the number of areas and their arrangement can be easily configured. The layout definition also specifies whether a series consisting of multiple images should be viewed in a stack or tiled manner.



Figure 7. Unlimited hanging protocols possible.

### 3.4 Additional Capabilities

In this section, we outline a couple of additional capabilities that a web-based PACS offers.

#### 3.4.1 Multiple User Sessions

There are no explicit limits to the number of user sessions that can be handled by a web server. Moreover, many sessions can view the same page. This feature can be used to show different aspects of a patient's studies to the Radiologist on several physical display monitors simultaneously. Figure 8 shows several sessions opened by a Radiologist to view the study, as in a traditional PACS environment. Another window can serve the purpose to dictate the report using the voice dictation and speech recognition capability is built into the device. In addition, web technologies enable a unique feature, collaboration, that is not found in traditional PACS. Along with viewing of images, users can initiate web-conferencing sessions and click-to-call with patients or other collaborators. As this is possible over a secure wireless network without any explicit need of VPN or LAN, the patient, referring physician or a sub-specialist can access the same study anytime, anywhere and on any device for instant web conference. Similarly, the current study can be very easily compared with a prior study or studies across different modalities that are stored on the database.

#### 3.4.2 Automatic Notifications on Abnormality Detection

Since the DICOM header tags are read by the web-based PACS when DICOM images are first loaded to the server, additional functionality can be integrated into the application. For example, Figure 9 demonstrates how a web-based PACS application can automatically send emails if abnormalities are detected when a DICOM image file is uploaded to the server.

## 4. SUMMARY AND CONCLUSION

In this presentation, a cloud-based system has been described that can display DICOM medical images on a standard web browser, also called a zero footprint viewer that requires no plug-ins, apps or specialized graphics drivers. The capabilities of these viewers rival those found on traditional PACS. Moreover, being a web-based application, additional capabilities such as automatic notification emails can be integrated into the workflow with relatively little ease.

## REFERENCES

- [1] W3C, [*Open Web Platform Milestone Achieved with HTML5 Recommendation*]. <http://www.w3.org/2014/10/html5-rec.html.en>.
- [2] Bolan, C., [*Cloud PACS and mobile apps reinvent radiology workflow*]. <http://www.appliedradiology.com/>.
- [3] Parikh, A. and Mehta, N., "PACS - next generation," *SPIE Conference Proceedings* (2015).
- [4] Pinykh, O. S., [*Digital Imaging and Communications in Medicine (DICOM)*]. Springer.

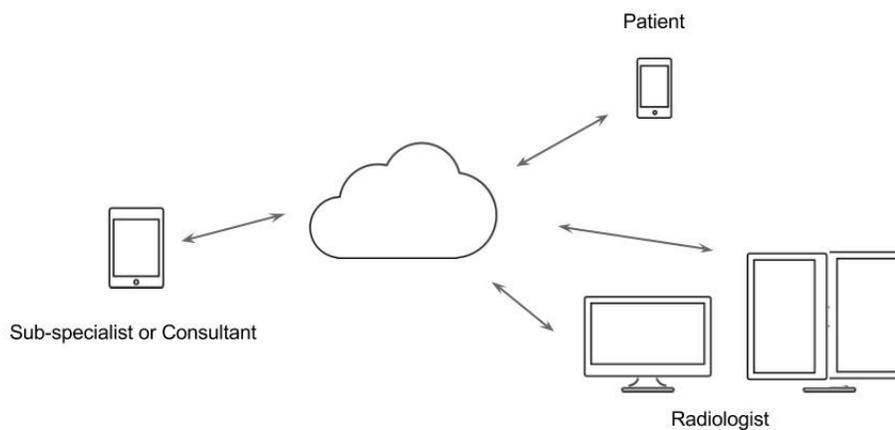


Figure 8. Several Simultaneous Sessions

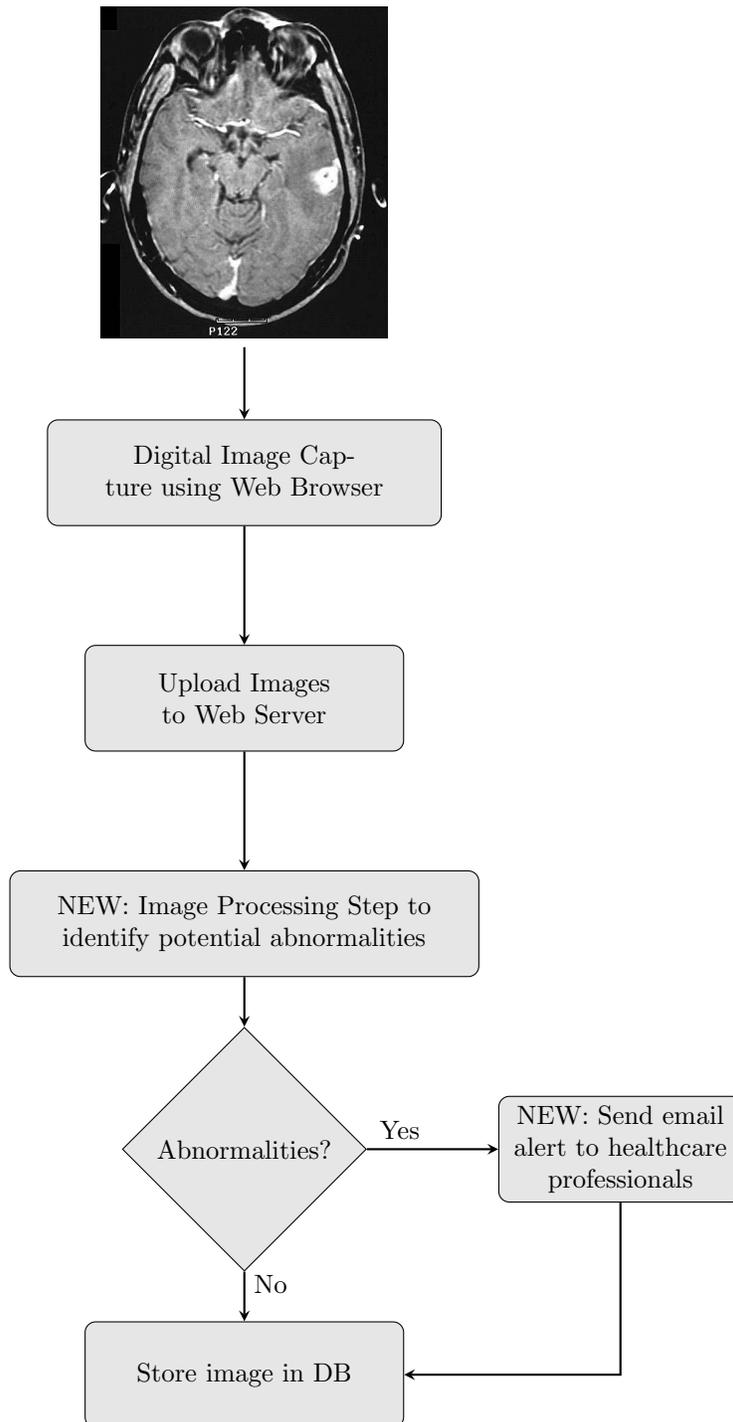


Figure 9. PACS - "Push" Model