Creating a Modern Electronic Medical Records (EMR) System

Jeremy Chaikind

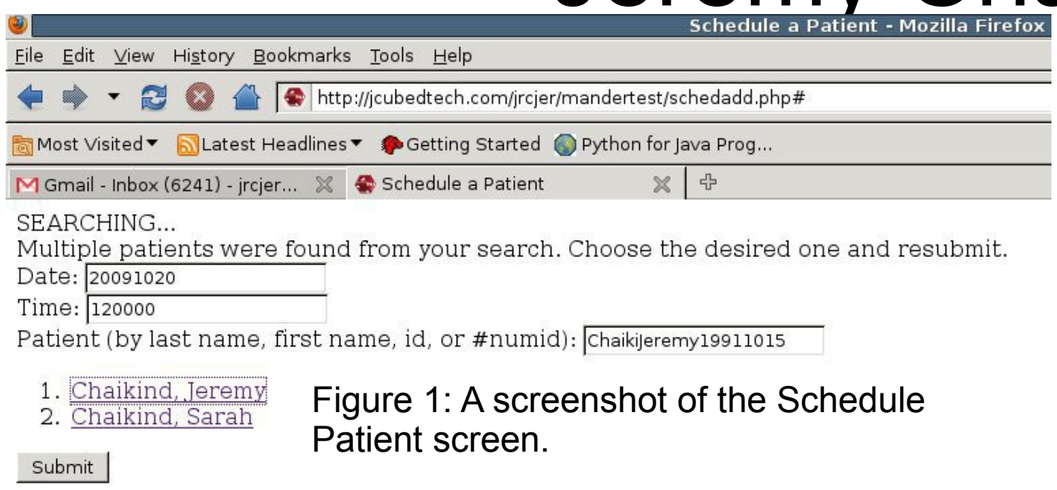

Figure 1: A screenshot of the Schedule Patient screen.

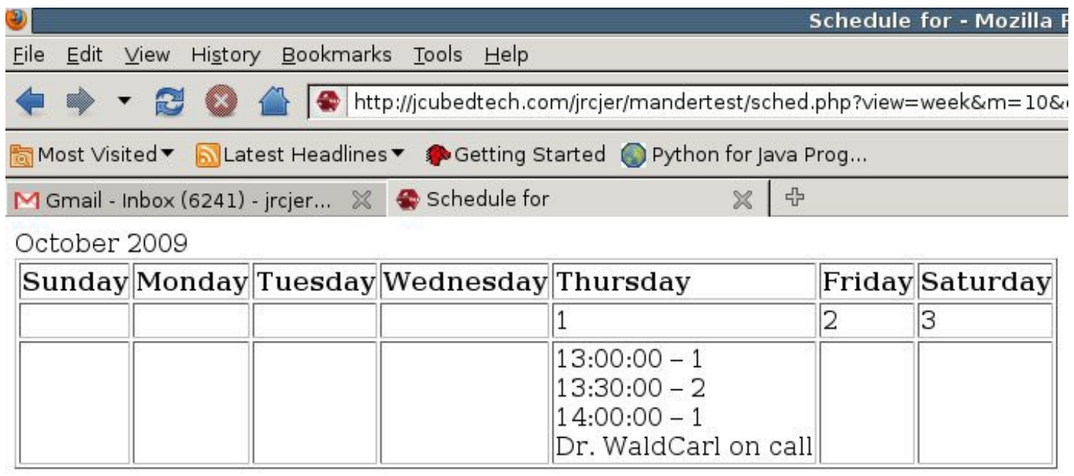

Figure 2: A screenshot of the Physician Schedule (week view) screen.

This project will attempt to create a functional user-friendly, HIPAA-compliant medical management and medical records (EMR) system. Web-based programing languages, such as PHP, HTML, and CSS will be used with MySQL databases. A variety of security measures will be explored in order to meet government standards (HIPAA) and protect patient privacy. Databases will be designed using the Relational Database Model and considering the ACID (*Atomicity*, *Consistency*, *Isolation*, and *Durability*) paradigm.

Introduction and Background

The business of medicine is a topic front and center for many Americans today. Beyond the question of health insurance reform, the United States government is in the process of changing the medical industry itself. Doctors have been given incentives to convert physical, paper charts to electronic ones in the near future. Soon after, physicians will be charged fees for using paper charts. Despite the exorbitant costs of many preexisting Electronic Medical Records (EMR) systems, some popular systems use older programming techniques and languages, and can be unintuitive and low-featured. This project plans to remedy the situation by creating an EMR system designed in conjunction with physicians to ensure ease-of-use, using forward-thinking web-based languages, including PHP, HTML, CSS, and MySQL.

Results

Work on the EMR system has largely been confined to exploratory work with PHP/MySQL setups. Basic tasks for EMR systems, including adding a patient, searching for a patient, scheduling a patient (Figure 1), and viewing a schedule by month or by week (Figure 2), were implemented. While little code from this experimental phase will be present in the final EMR system, implementing EMR screens allowed MySQL proficiency and understanding to be cultivated while prototypes for the final screens were considered. Because these constructions of EMR tasks were not designed to be integrated into the final project itself, unstylized HTML forms were used for the practice screens.

Discussion

Although much of this initial code could not be salvaged for the final EMR system itself, the work done
in this first quarter was worthwhile for a number of reasons:

- Programming in a web-based environment reinforced syntax, techniques, and proper coding practices of the internet.
- Programming with PHP and MySQL helped to expand the researcher's knowledge and understanding of both languages and the interfaces between them.
- Despite ignoring style and UI tweaks, creating practice medical *screens* enabled fundamental structures of the user interface and the program itself to be considered