Creating a Modern Electronic Medical Records (EMR) System

Jeremy Chaikind

Adding a new chart (unstylized) in Mander.

A patient's "facesheet" (unstylized), exhibiting his identifying information and listing his charts.
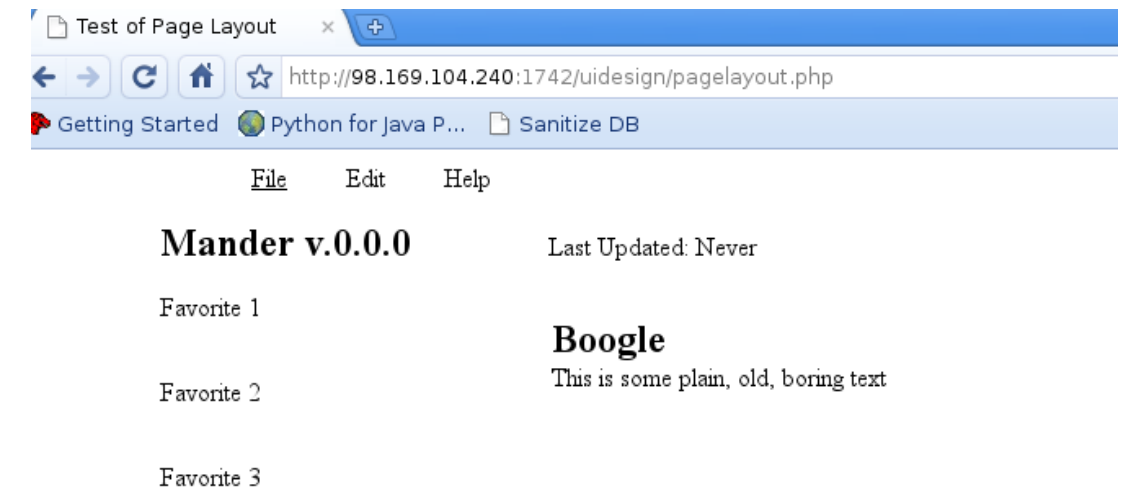
Initial experiments in UI – positioning.

Abstract

This project will attempt to create a functional, user-friendly medical management and medical records (EMR) system. Web-based programming languages, such as PHP, HTML, and CSS will be used with MySQL databases. Databases will be designed using the Relational Database Model and considering the ACID (Atomicity, Consistency, Isolation, and Durability) paradigm. The EMR design will ensure expandability, intuitive interface, and practicality for the end user.

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

Second quarter work on the EMR system included starting all aspects of program design. Much of this quarter was spent in the design and early implementation phases of the final, "Mander" EMR system. First, a personal LAMP (Linux, Apache, MySQL, and PHP) server was created. Next, the final database design was determined. Based on the Relational Database model, a global user ID was implemented to connect an abstract *person* to all of his/her data across many different tables. The Relational model also permits a *person* to have multiple rows in a table (for instance, multiple phone numbers). After the new database design was completed, experiments began in user-designed templates, first using a WYSIWYG editor (OpenWYSIWYG, licensed under a Lesser General Public License) and then an RTF import. However, neither option was feasible, due to complexity of implementation and difficulties with the user interfaces. Finally, basic steps in interface design were achieved, working on menu bar design and general positioning of elements.

Discussion

First and Second Quarter work, while relatively unexciting, serves as necessary groundwork for this project. First Quarter work was primarily used as an experiment in PHP/MySQL development and initial prototypes for database organization. Second Quarter work was much more important to the final application itself, creating the LAMP server framework upon which the rest of this project will be based. The creation of a library of database functions will be extremely useful for page-based database interactions, especially those generated by end-user action (i.e. those used to connect user templates to the database). Experimentation in template management served as a first step into finding a method to allow the Mander EMR to become fully expandable, greatly increasing its applicability and lifetime. Finally, basic forays into user interface design will provide a backbone for a final user interface for this program that will make it user-friendly enough for medical personnel to have little difficulty converting to the new system.