Next Gen Scheduling
Using MongoDB and AngularJS

Abstract
The purpose of this project was to develop a new web-based scheduling system that improved upon Aprima’s existing, non-web solution. We build on top of the last team’s database research to develop a complete system that allows for medical providers to schedule appointments for patients. The web interface was developed in Angular-JS, while the backend was developed with ASP.NET, which fits in with Aprima’s overall development direction. Our solution provides Aprima with a base for them to expand upon as they transition their existing products to a web-based platform. Furthermore, our solution provided Aprima with data on MongoDB’s performance in a real application environment, which further supported their decision to move their existing relational SQL solution, to a nonrelational MongoDB backend for future applications.

Results
Main Page/ Calendar Page

Architecture

User visits Web App
User sees Scheduling App

Front-End
Built Using
AngularJS
JavaScript
HTML
CSS

Middle-Ware
Built Using
C#
ASP.net
Query DB for Data
Consume Data & Create Data Object

Back-End
Built Using
Mongo DB
Retrieve Data
Send Data to Back-End

Metrics
For the project guidelines Aprima wanted 4 main functionality for the application. These included: Add, Delete, Merge, Modify. In addition to this, Aprima wanted OAuth 2.0 for authentication of users. By Aprima’s recommendation, we decided to not implement OAuth for the application. Overall, we met the project requirements and Aprima was satisfied with the results.

Impact
Our project will help provide a basis for a newly updated scheduling software for appointment booking. The project has a color-coded calendar view that allows the user to, at a glance, understand what appointments are expected for the day and the rest of the week. This allows for an increase in usability for the user. As part of Aprima’s goals to move from a relational to non-relational database, our project lays the groundwork for development using MongoDB instead of a relational database. This project was to test the feasibility of MongoDB as opposed to other options. This project shows the efficiency of MongoDB with extremely low database lookup times.

Summary
Aprima Medical solutions had two primary problems they looked to address. Firstly, their existing solution was a native application, which limited their ability to rapidly expand their product to different types of devices other than Windows. Secondly, their current application used Microsoft SQL, which was suffering from performance issues due to the large amount of stored data. Our application provides solutions to these two issues. We developed a new web-based application with Angular-JS, which provides Aprima with some ideas on how to transition their other existing products to the web. Last semester’s research led us to implement MongoDB for the backend database due to its reliability and performance. As a result, Aprima can benchmark the improvements the new nonrelational database system provides in a real application setting.