



Design Specification

Libraries We Will Be Using

- `express.js` - Organize routes and views
- `postgres.js` - Access SQL database
- `D3` - Model the course list views
- `jQuery` - Manipulate the DOM
- `AJAX` - Dynamic page manipulation
- `Morgan` - User login functionality

• Birds-eye View

- Login --> Main
 - The student logs in and is taken to a page where he/she sees a table showing the courses taken so far, currently, and planned to take in the future. There are also navigation tools for displaying general requirements, class search, and current class schedule. This interacts with the server by pulling the appropriate data for the user that is signed in, in the form of JSON objects. Except for times of course enrollment, the student is mostly pulling data from the server and not pushing anything to the server

Birds-Eye View (Cont.)

- Login --> Admin
 - The admin, on the other hand, has the power to adjust things and push data to the server and database. When an admin logs on her/she can see an entire list of courses offered and be able to make adjustments to any field, which update on the server (not dynamically, there will be a "submit changes" button). The admin can also pull and adjust student data for logging grades, fixing errors, etc.

Components

- Database
 - Functionality: store students' information (username, password, student ID, course history), existing classes' information (prerequisites, term, instructor). Only admins can modify content of the database (add/remove classes, create users).
 - Implementation: javascript module that includes methods to access the database:
 - Add/remove/customized search user
 - Add/remove/customized search classes

Components (Cont.)

- **Server Routes**
 - **Functionality:** connect pages. It specifies the rendered view for each route and direct it to other routes when certain events are triggered by the users or admins
 - **Implementation:**
 - **Login:** route to either user main or admin main depending on which account is logging in
 - **User main:** route to pages and actions available to the user: calendar, course list, logout.
 - **Admin main:** route to pages and actions available to the admin: modify the database, view courses.

Components (Cont.)

- Client Views
 - User begins at a login page. After logging in the user is taken to the main page, at which point he/she can look at class schedules, GE requirements, degree paths, etc. Admin users will instead be taken to a page that allows them to view and update course offerings.
 - Views will be .ejs files and navigation will be organised using express. Details such as drop down menus and other DOM manipulation will be handled in JQuery. We will use AJAX for features like adding classes to your "shopping cart" which require server requests without re-rendering the page. Visualization of degree paths and class schedules will be done using D3.js

Challenges

- We have little experience with UI's and making it all look good could prove difficult
- AJAX requests to populate shopping cart might be a pain to figure out
- Avoiding procrastination
- Making everything extendable and dynamically updateable
- Getting data for all the courses and tracks
- How to make students and their data easy to keep track of and store
- Dividing up work evenly and making it all work together on time
- Integrate jquery with ejs

Who is Doing What

- Sam: Server
- Yong: Database
- Josh: Routes
- Drew: Front-end views
- Alex: Front-end functionality
- Khanh: Graphic design

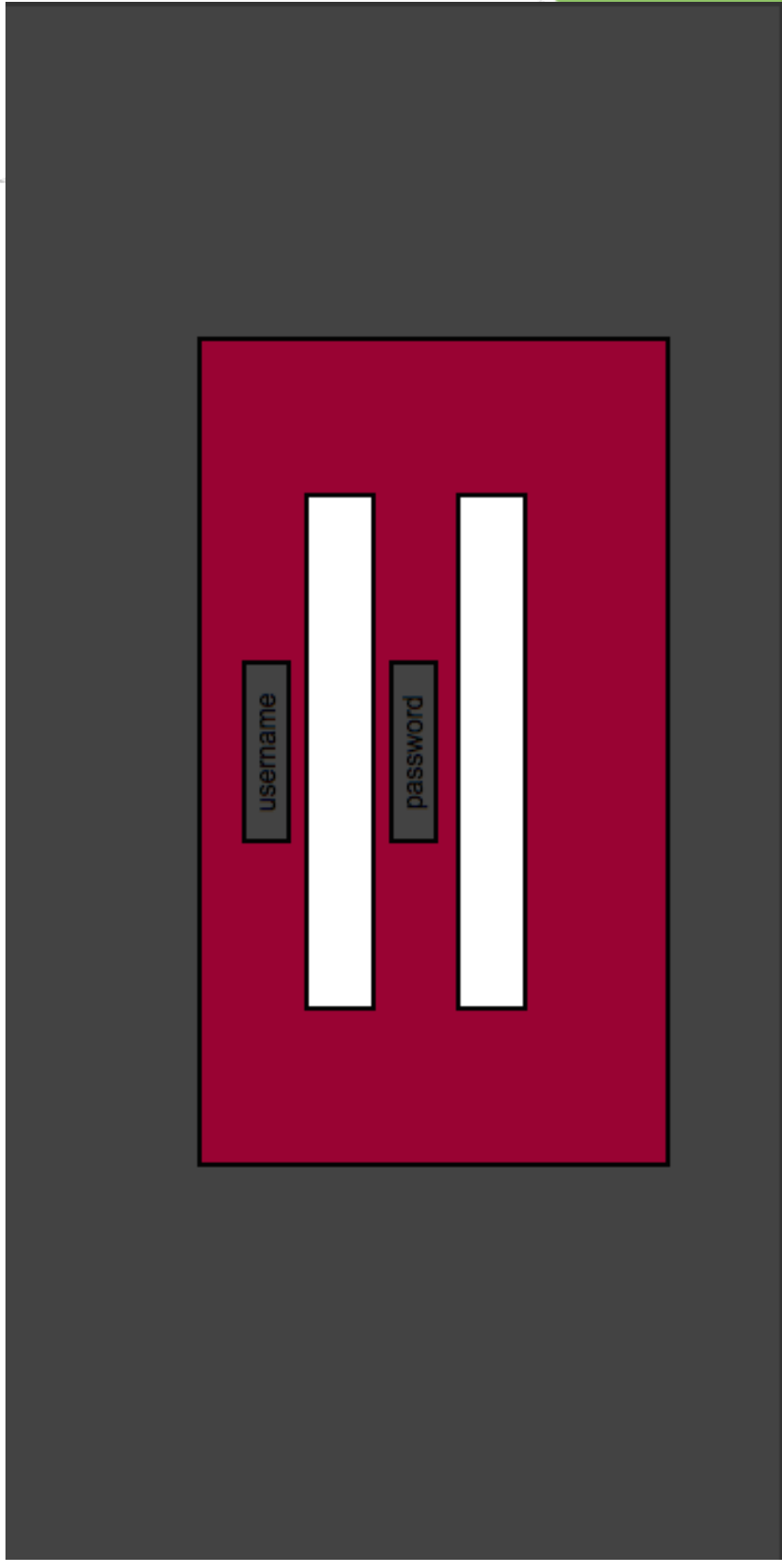
Views to Work on

- Login page
- Student main page
- Admin main page
- Admin student info page
- Student course list page (displays courses, when they are offered, etc)
 - Course description lightbox

Views to work on (Cont.)

- Gen ed requirements page
- User settings page
- Course enrollment page(s) (if time)
 - query for classes page
 - shopping cart/enrolled/suggested page

Login Page View



A diagram of a login page layout. It features a dark gray rectangular background. Centered on this background is a red rectangle. Inside the red rectangle, there are two white rectangular input fields stacked vertically. To the left of the top input field is a small dark gray box containing the text 'username'. To the left of the bottom input field is a small dark gray box containing the text 'password'.

Main Page View

my classes		gen ed requirements				courses		sign out
semester 1		semester 2		semester 3		semester 4		semester 6
course taken		course taken		course taken		find a course		find a course
course taken		course taken		course taken		find a course		find a course
course taken		course taken		course taken		find a course		find a course

Admin Main Page

X					
Class Name	Course Number	Prerequisites	Semesters Offered	Students Enrolled	Capacity
CMPSCI110	17623988	N/A	Fall, Spring	25	25
CMPSCI130	27836920	N/A	Fall	77	100
CMPSCI187	78921609	N/A	Fall, Spring, Summer	12	50
CMPSCI220	90281726	CMPSCI187	Fall, Spring	23	25
CMPSCI230	73222783	CMPSCI187	Spring	2	50
CMPSCI240	10011928	CMPSCI187	Fall, Spring, Summer	27	50
CMPSCI250	29001223	CMPSCI187	Fall	20	25

Calendar View

		Semester: Fall 2014			Click Blank Tiles to Add Comment	
		Monday	Tuesday	Wednesday	Thursday	Friday
Day Time Block						
8:00 AM			CMPSCL-220-L 8:30-9:45 Herter Hall 227		CMPSCL-220-L 8:30-9:45 Herter Hall 227	
9:00 AM						
10:00 AM						
11:00 AM						
12:00 PM				CMPSCL-250-D 12:20-1:10 Engineering Lab 323		
1:00 PM		CMPSCL-250-L Hasbrouck Add 126		CMPSCL-250-L Hasbrouck Add 126		CMPSCL-250-L Hasbrouck Add 126
2:00 PM						
3:00 PM						
4:00 PM			CMPSCL-326-L Goessmann 64		CMPSCL-326-L Goessmann 64	
5:00 PM						
6:00 PM						

