Course Planner Design Document

1. Introduction

Course Planner is to be a website that will enable students to efficiently and easily schedule their coursework and UBC courses together in one place. Users will be able to receive daily reminders informing them of the tasks they have to complete per day as well as register in UBC courses and have a course calendar automatically created for them that they can then complete with any custom, daily tasks they wish to add. Course Planner is unique to other existing scheduling applications in that it integrates directly with UBC's course database and allows users to receive reminders not only for their courses but also for customized daily tasks.

Core features:

A. Course Schedule
   1. Grab all courses information from UBC website into our database
   2. Add courses which are inside our database

B. Calendar System
   1. View course and task schedule
   2. Add//Delete existing calendar entries (course/task)

C. Reminder System
   1. Receive e-mail reminders
   2. View Daily reminders on-site

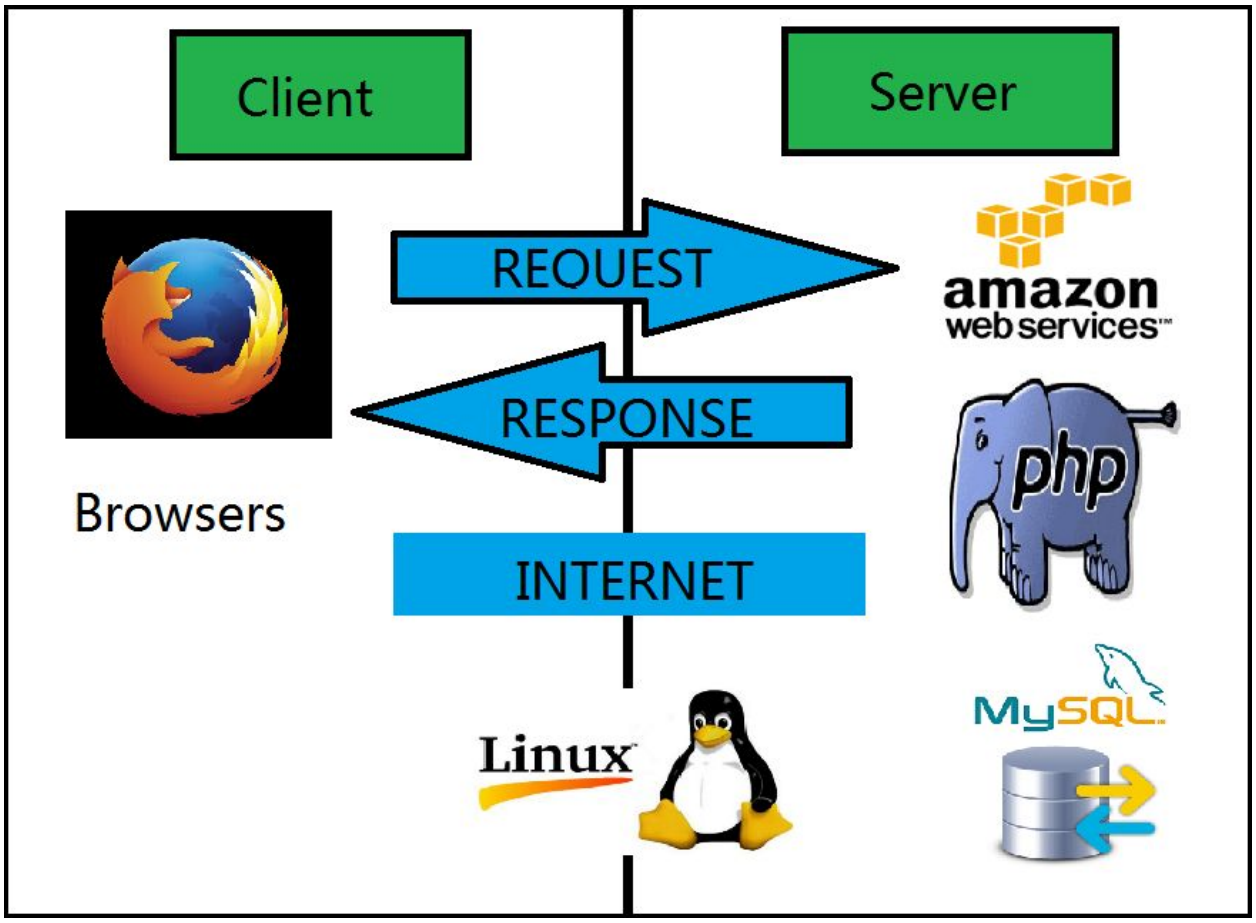
Additional features:

a) Create or login to user profile
   1) Use facebook to login to Course Planner profile
   2) Create a new course planner profile after first login in with Facebook

2. Architectural & Component-Level Design

LAMP

The web server is written in LAMP which is is an archetypal model of web service solution stacks, named as an acronym of the names of its original four open-source components: the Linux operating system, the Apache HTTP Server, the MySQL relational database management system, and the PHP programming language. LAMP is a suitable solution stack for building dynamic web sites and web applications. The model view is as follow image.



Linux operating system

Linux is a Unix-like and mostly POSIX-compliant computer operating system (OS) assembled under the model of free and open-source software development and distribution. The defining component of Linux is the Linux kernel, an operating system kernel first released on September 17, 1991 by Linus Torvalds. The Free Software Foundation uses the name GNU/Linux to describe the operating system, which has led to some controversy. Source: https://en.wikipedia.org/wiki/Linux

LAMP is required to use Linux operating system to manage, edit and modify our web application.

Amazon web services

AWS a subsidiary of Amazon.com, offers a suite of cloud-computing services that make up an on-demand computing platform. These services operate from 13 geographical regions across the world. The most central and best-known of these services arguably include Amazon Elastic Compute Cloud, also known as "EC2", and Amazon Simple Storage Service, also known as "S3". As of 2016 AWS has more than 70 services, spanning a wide range, including compute, storage, networking, database, analytics, application services, deployment,

management, mobile, developer tools and tools for the Internet of things. Amazon markets AWS as a service to provide large computing capacity quicker and cheaper than a client company building an actual physical server farm. Source: https://en.wikipedia.org/wiki/Amazon_Web_Services

We have chosen Amazon web services as our application's web server, because it is free and provides the LAMP platform for building web applications. All the related web application's files will push up to AWS for running the actual web application.

MySQL

MySQL is an open-source relational database management system (RDBMS). Its name is a combination of "My", the name of co-founder Michael Widenius' daughter, and "SQL", the abbreviation for Structured Query Language. The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. For proprietary use, several paid editions are available, and offer additional functionality. MySQL is a central component of the LAMP open-source web application software stack. Source: https://en.wikipedia.org/wiki/MySQL

PHP programing language

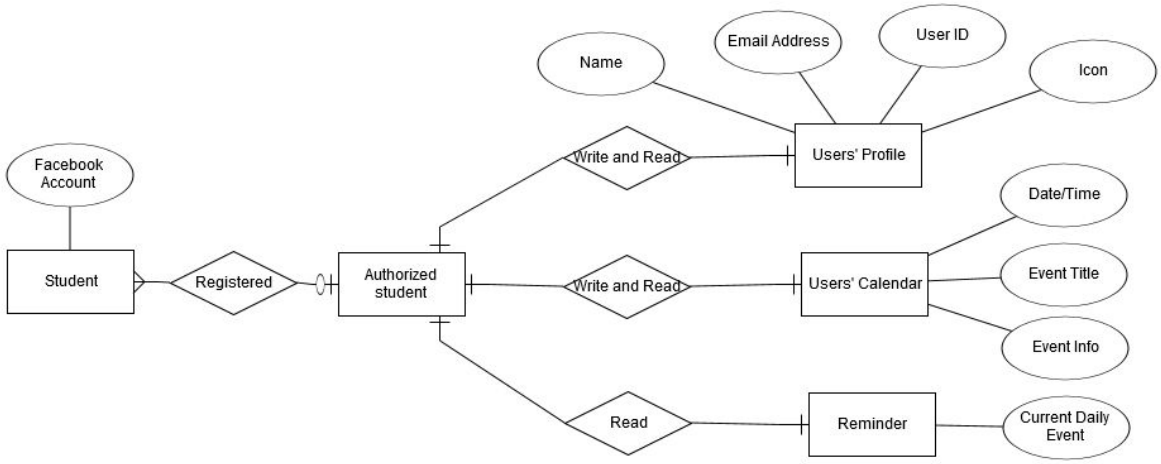
PHP is a server-side scripting language designed primarily for web development but is also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Development Team. PHP originally stood for Personal Home Page, but it now stands for the recursive acronym PHP: Hypertext Preprocessor. Source: https://en.wikipedia.org/wiki/PHP

Javascript

JavaScript is a high-level, dynamic, untyped, and interpreted programming language. It has been standardized in the ECMAScript language specification. Alongside HTML and CSS, JavaScript is one of the three core technologies of World Wide Web content production; the majority of websites employ it, and all modern Web browsers support it without the need for plug-ins. JavaScript is prototype-based with first-class functions, making it a multi-paradigm language, supporting object-oriented, imperative, and functional programming styles. Source:https://en.wikipedia.org/wiki/JavaScript

3. Data

Our Web Server Entity-Relationship (ER) Diagram is as follows:

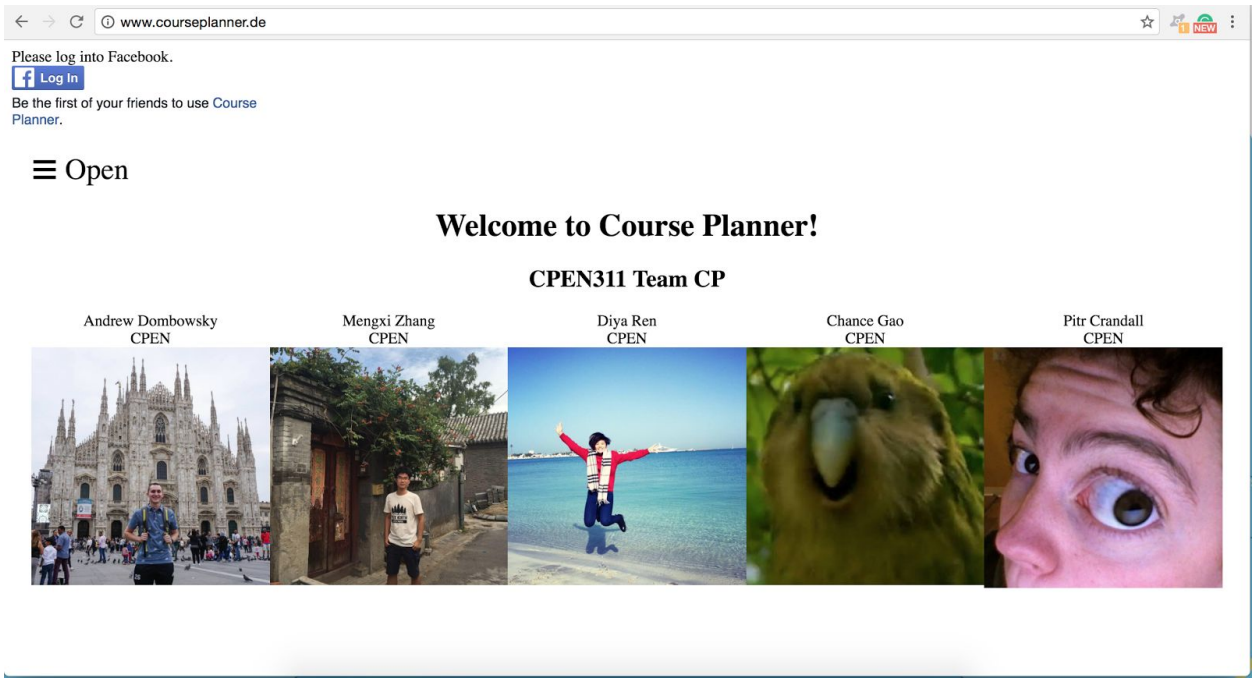


UBC students need to register a Website account to access the calendar system, reminder system and users' profile system. The user's' profile system will have the user's information and email address. The calendar system will display the related course schedule and info and the user's' personal schedule. The reminder system will be used to remind the current daily events based on the calendar system. All users are merely have one profile page, one user's calendar page and one reminder page.

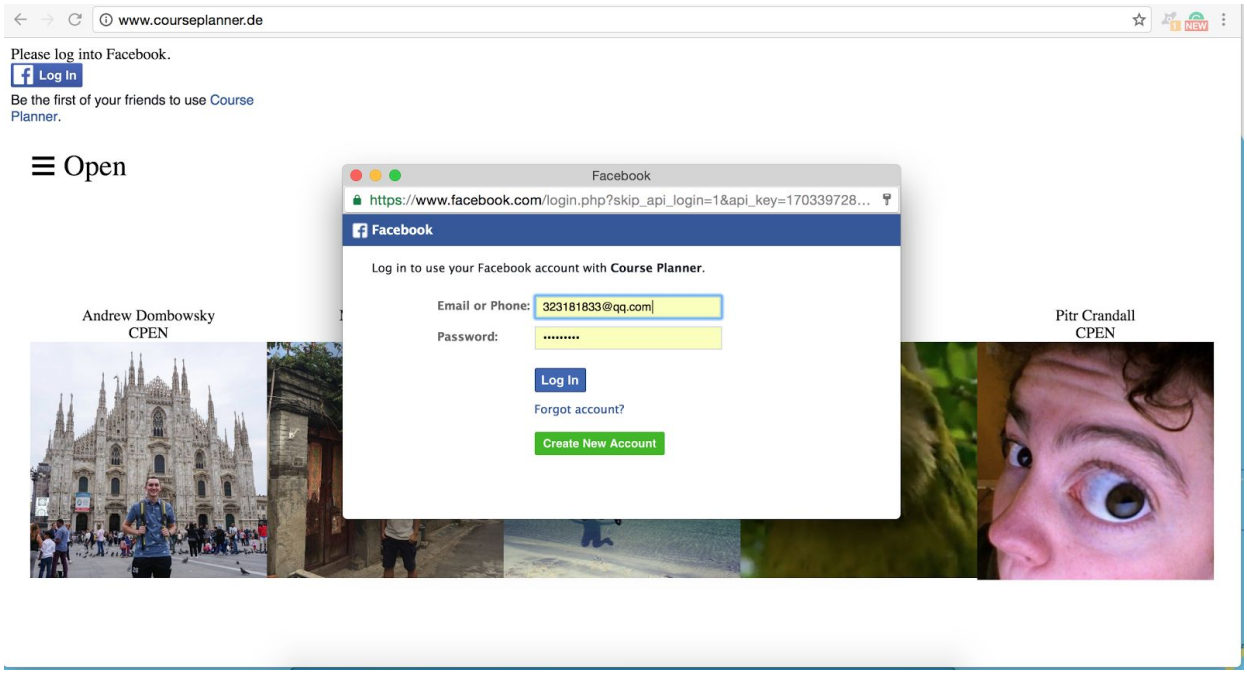
4. User Interface Design

The user interface mainly includes the login view, profile modification panel, reminder panel, and contact and help panel. The login view is used to allow users to login with their facebook account, which requires a facebook login module to support the function. After the user logs into their account, they will be able to navigate through all the following four panels on a sidebar. Users could open or hide the sidebar by clicking opening button or close button. The panels will display user's private information related to their account, and users could modify their schedule, view reminders and get contact and help from our design team on corresponding panels. All the panels are able to access certain parts of our server end database to retrieve or update information. Also, there should be a reminder module to support its functionality.

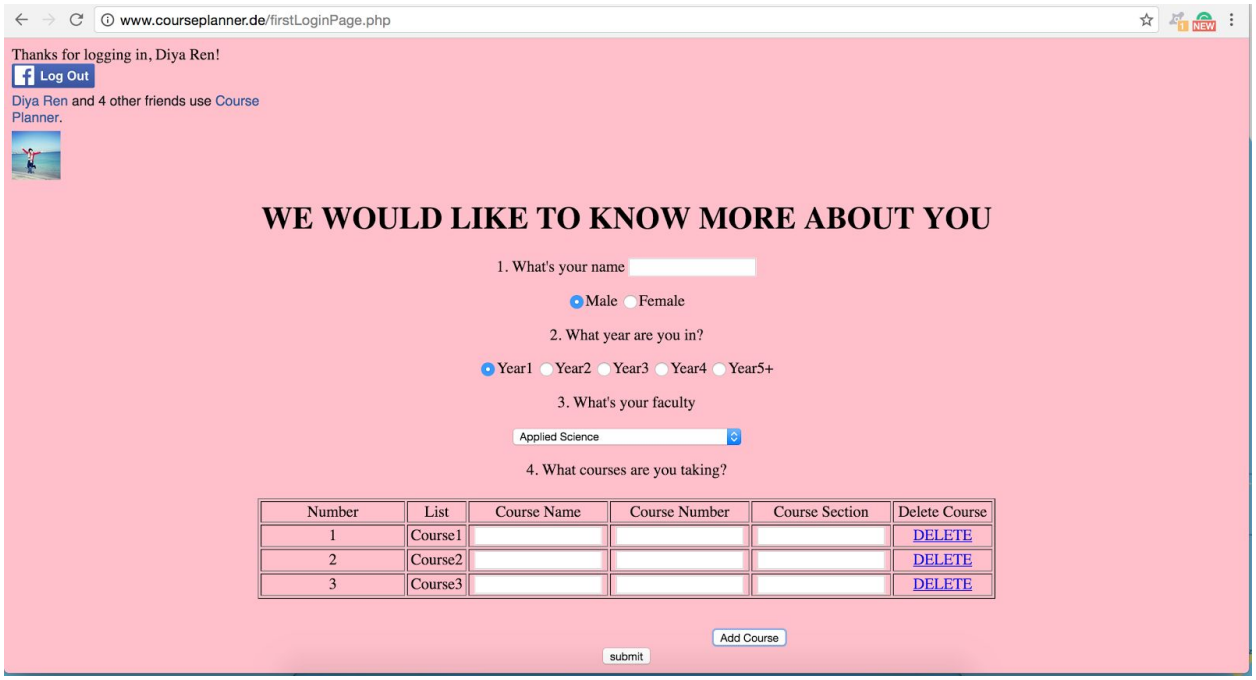
a)Dashboard View:



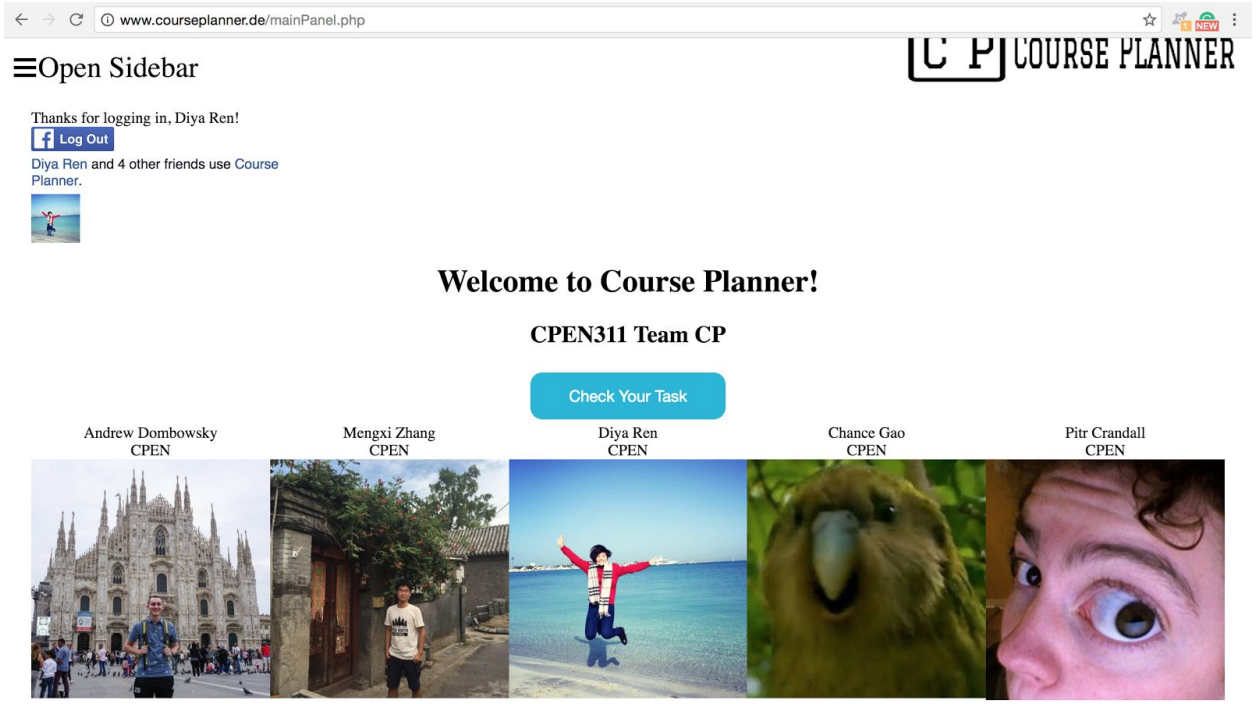
b)Facebook Login View:



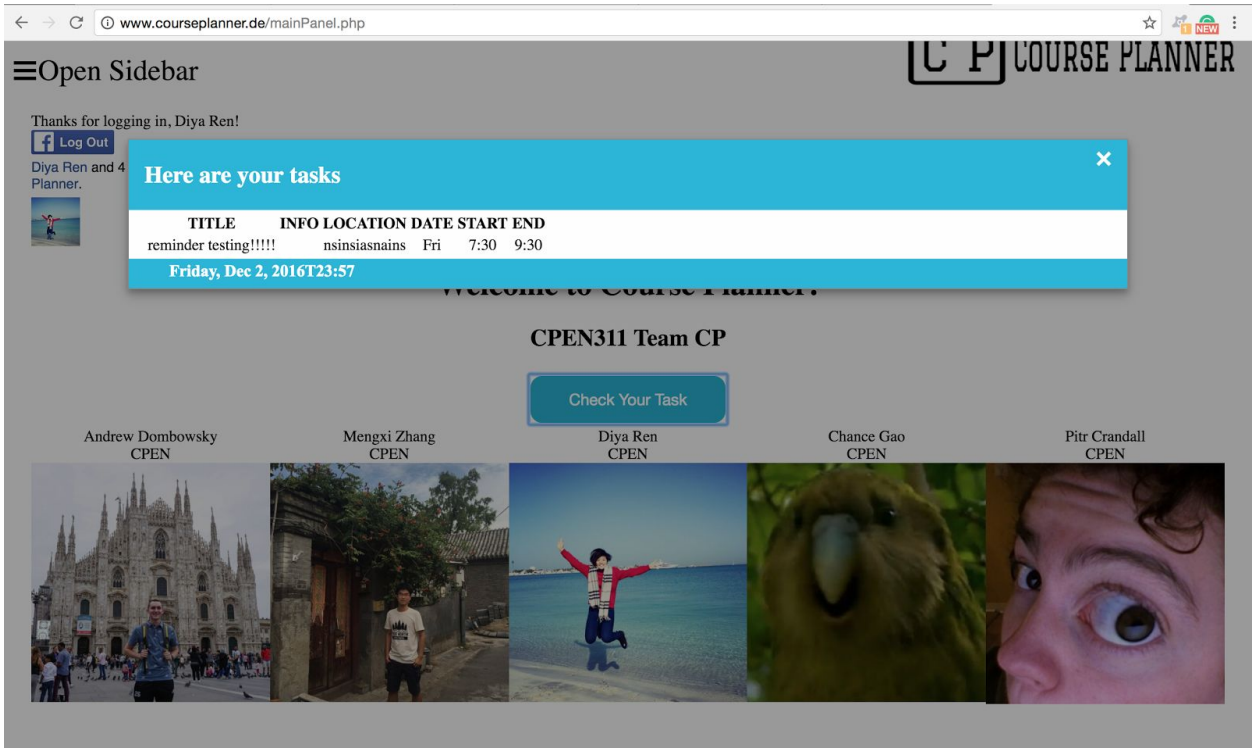
c)First login Info Enter Page View:



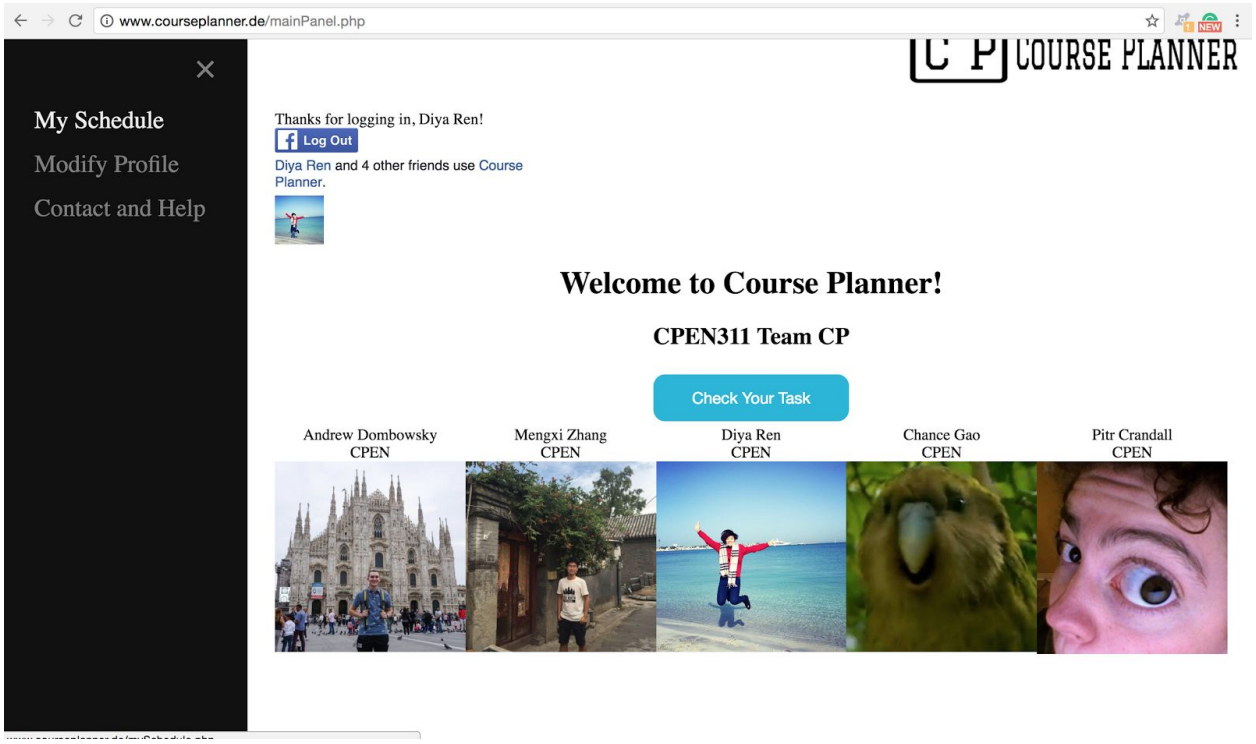
d) Check Your Task Button View



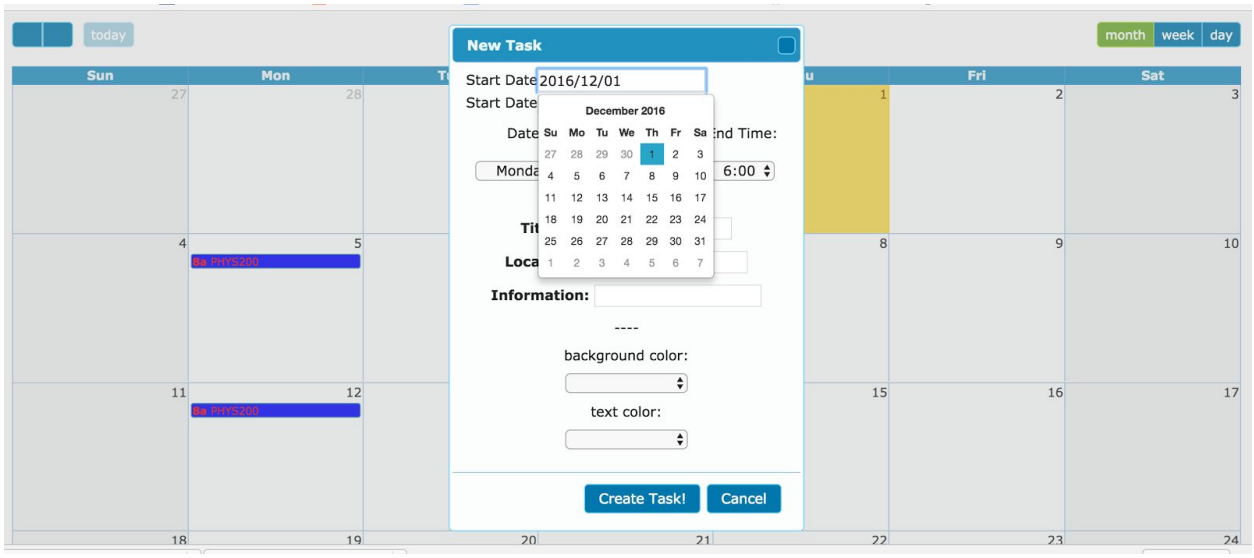
e) Tasks Display View

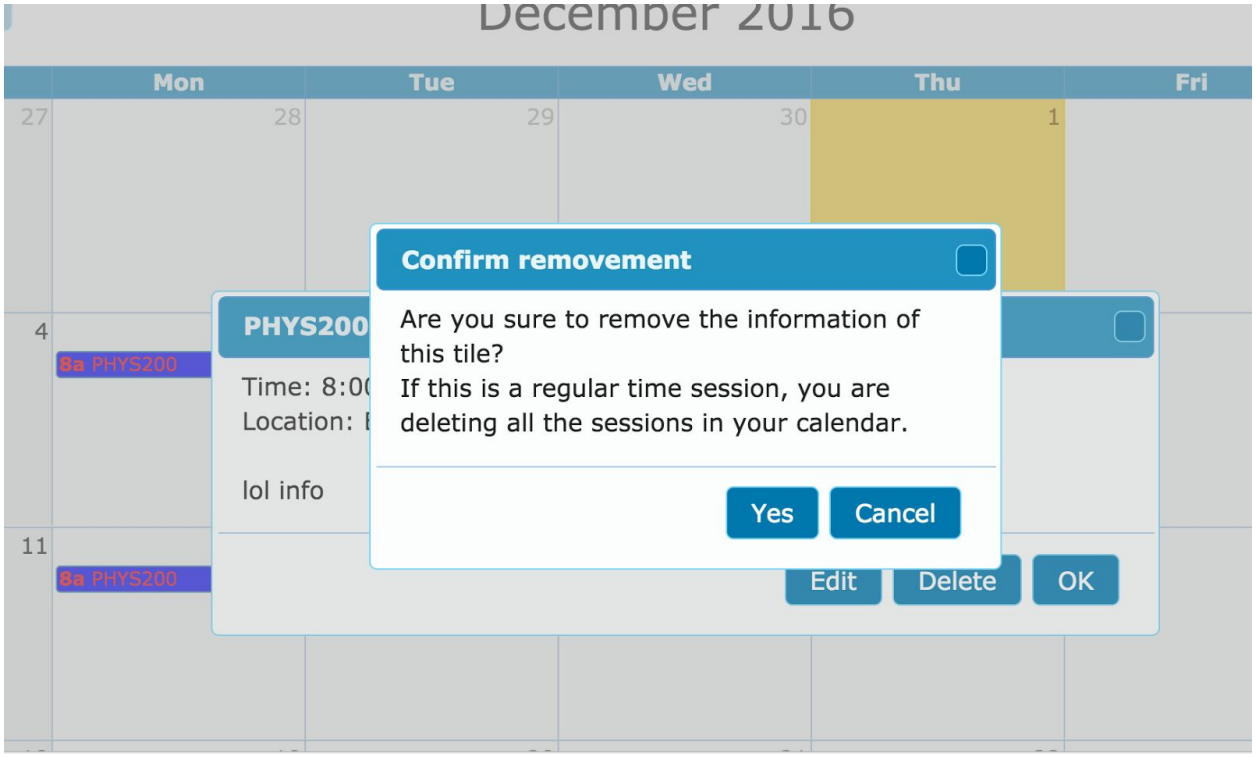


f)Sidebar Select View:



g) Calendar Information View:





5.Restrictions, limitations, and constraints

The most important constraint will be the scope of this project. The scale of the client right now is limited to students who are attending UBC. In the future, we may want our project to be able to be used widely in different universities. As the number of users and the scope of project increase, it will likely require more time to maintain the website and more money support on this project in order to keep it running functionally.

6.GUI

The user interface is built using HTML, CSS, PHP and Javascript libraries.

7.Validation

The UI validation will be a preceding process with the involvement of the clients. The design progress is based on a weekly basis and client feedback taken into account to design the user interface as user friendly as possible in order to ensures that the product actually meets the user's needs, and that the specifications were correct in the first place.

8. Installation Guide

MySQL Database Installation Guide:

The website used the MySQL database provided by Amazon Relational Database Service (RDS). It can be connected by using any standard SQL client application to connect to a database on the DB instance. RDS is easy to set up, operate, and scale a relational database in the cloud. It provides cost-efficient and resizable capacity while managing time-consuming database management tasks. In the project development process, the PhpMyAdmin online software is used to connect, modified and update the MySQL database. To use the PhpMyAdmin software, it merely need to install the PhpMyAdmin software in Amazon EC2 instance, which in our case is an Amazon Linux instance, so installations are performed using yum.

1. Enable the RPMforge repository on a Linux system as PhpMyAdmin is not available in the official Linux AMI repository. The following command is used: rpm --import http://dag.wieers.com/rpm/packages/RPM-GPG-KEY.dag.txt
2. Install PhpMyAdmin as the following command: yum install PhpMyAdmin
3. The Apache configuration need to be changed, so that PhpMyAdmin allows connections not just from local host. Using command: vi /etc/httpd/conf.d/phpmyadmin.conf
4. The authentication in PhpMyAdmin need to be changed from cookie to http: and host from local to RDS host URL. Using command: vi /usr/share/phpmyadmin/config.inc.php
5. Restart the Apache service.
6. The RDS database can be using http://<Domain name>/phpmyadmin to access.

The advantages of using PhpMyadmin over command line interface are:

1. The queries within the MySQL can be directly run.
2. It can check referential integrity in MyISAM tables.
3. All SQL-statement and batch-queries can be executed, edited and bookmarked.
4. Multiple MySQL servers can be managed by PhpMyAdmin.
5. Various formats can be used: CSV, SQL, XML etc.
6. Using Query-By-Example(QBE), create complex queries automatically connecting required tables.
7. It supports InnoDB tables and foreign keys.

Apache Web Server Installation Guide:

The website runs on a LAMP software stack and therefore one of its main components is the Apache web server. The steps in setting up an installing the website's source code to the server are as follows.

1. Install php, apache and mysql via the following commands: sudo yum install -y httpd24 php56 mysql55-server php56-mysqlnd
2. Start the web server: sudo service httpd start , and set it to configure at each system boot: sudo chkconfig httpd on
3. Copy the site's files from your local system to the remote server's webroot folder via an scp command: scp /path/to/local/files username@publicServerDNS:/var/www/html

Changes and Rationale

One significant change made to the design is due to relaxing our requirement to distinguish between students and TAs/professors. This exclusion only affects our database. Our database is now simpler than it was before and excludes some functions from needing to be implemented on the server. All assignments and exams will now be added and ranked by students, as this now our only type of user. We will no longer be showing any user profile information that is not directly pertinent to said user; only their name, contact information etc.

We will no longer be including the ability to create, rank and edit course features. This is because, based on the stage we are currently in with development, we believe that we should focus on improving the current implemented features to work at full and proper capacity by the demo day. Since administrator accounts are also no longer under our consideration, the ranking system's removal will not affect that feature. Additionally we will be simplifying our reminder system to send reminders for all tasks and courses in a user's calendar on a day-by-day basis and not only for specific course features such as exams and assignments.