SCHOOL OF
COMPUTER SCIENCE

THE UNIVERSITY
of ADELAIDE

COMP SCI 2207 – Web & Database Computing

Group Practical

Aims:

1) Understand principles of designing user friendly websites.
2) Write standards compliant websites in HTML & CSS.
3) Develop dynamic, client-side web content using Javascript.
4) Use third party APIs in web applications.
5) Understand the interactions between the client-side and server-side components of web applications.
6) Design and build AJAX calls to a server and handle responses.
7) Use OpenID to provide secure authentication services.
8) Use E-R Modeling to design a database structure.
9) Understand the role of databases in web applications.
10) Plan & integrate a database into a web application.

References needed:

- Notes from lectures.
- Notes and code from practical exercises.

Overview:

Your group assignment for the Semester is to design and develop a personal journal web application that allows users to make a journal about their day based around the Google Calendar Events from that day.

This assignment should be worked on in groups of 3. You will need to form a group prior to completing and submitting part 1. Your group must all be enrolled in the same Tutorial session. If you are not already part of a group, use the MyUni Discussions page to find other group members. If you are unable to form a group of 3 students from your tutorial session, contact the lecturer.

Note that, although the other content of this course will cover the general principals needed to develop web applications, you will likely need to look up specific methods, properties and events in the appropriate documentation and APIs to complete this web application. If you're unsure exactly what's needed to complete a task, you should attempt to find answers yourself before requesting help. If you ask supervisors, they will focus on showing you how to find them.

Assessment:

Part 1 – Designing your website *Due at the beginning of your Week 3 Tutorial*	5%
Part 2 – Building your website *Due at the beginning of your Week 5 Tutorial*	5%
Part 3 – Google Calendar & setting up your server *Due at the beginning of your Week 7 Tutorial*	5%
Part 4 – Client-server integration *Due at the beginning of your Week 9 Tutorial*	5%
Part 5 – Adding database storage *Due at the beginning of your Week 11 Tutorial*	5%
Final Presentation – Show us what you got! *Held during your Week 12 Practical session*	5%
TOTAL	**30%**

As written above, this group practical is worth a total of 30% of your final course grade and is NOT a hurdle requirement. Each of the 5 components and the Final Presentation contribute 5% to this total. A detailed mark breakdown will be provided for each part shortly.

Part 1 – Designing your website

Before you begin:

- ✓ Read the overview above.
- ✓ Form a group.

Task:

Think about and explore web pages. What might a web site that allows you to write a journal look like? What information might users be required to supply? Will they need to log in? We know we'll be using Google Calendar, but how would Events integrate with the journal notes? Should a use be able to write notes about their day even if they don't have any calendar entries for that day? You are encouraged to think about additional features for this application; we have outlined the minimum.

- ✓ As you explore web pages, record any pages you find that are good examples of what your site could look like. What parts work well. What parts don't work well?
- ✓ Also record any examples you find of features you might want to include in your site.

Once you decide what a user will be able to do, then you need to think about how they will do each task and how you will make those tasks visible on your site – keeping in mind the articles and lectures on design and usability.

- ✓ Draw (don't code) what your site will look like.
  - ○ It should be obvious from the drawing what each part of the site does.
  - ○ Be sure to include any dimensions – if the window is resized, how do elements change?
- ✓ Describe how the features of your site will function.
  - ○ What interactions are available to the user?
  - ○ What happens when a user performs one of these interactions?
  - ○ If a change occurs to the page/site following an interaction, clarify what that will be and draw if necessary?

Once you have a basic design, you now need to review and tweak it for usability and accessibility.

- ✓ Working with another group or individuals outside of your group, perform basic usability testing.
  - ○ Record what you did to perform usability testing and any findings.
- ✓ Review your design.
  - ○ How does it minimise kinetic and cognitive load?
  - ○ Does it meet the standards & heuristics references in lectures and on MyUni?
- ✓ Record any updates to your design made as a result of the above testing and review.

Notes:

As you learn more about implementing web applications throughout this course, your design will likely change; that's okay, but be sure that as those changes creep in, you continue to review your site for usability and accessibility. We will be assessing you for this in your final presentation.

Submitting your work:

You will be presenting your work with your group for review by a tutor during your tutorial session.

- Be sure to have your work ready **at the start of your tutorial session** in **either hard or soft copy format**.
- Submit your work in a ZIP file to the link on the course website before the end of your session.
- Your tutor review your work and ask questions before assigning a mark.
- Your mark will be available in the course website's gradebook before your next session unless otherwise noted.

Part 2 – Building your website

Before you begin:

- ✓ Review practical exercises 1-4

Task:

Now that you have a solid design, it's time to start coding it!

- ✓ Using the basic HTML and CSS that you've learned thus far, write code to implement the layout of your main page.

Continue to develop and implement your design in HTML and CSS. You should now have a firm grasp on these languages and be able to implement ALL of the static elements of your website.

- ✓ Use HTML and CSS to implement all of the pages in your site.
- ✓ Follow best practices as outlined in lectures and exercises.
  - ○ Ensure CSS is used appropriately to provide a consistent and maintainable style across all your pages.
  - ○ Ensure your implementation passes validation.

As you build your website, begin to add interactive features using JavaScript and jQuery.

- ✓ Use JavaScript and jQuery to further build your design.
- ✓ Add the ability for users to make entries and notes about their day.
  - ○ Notes made should show up after a user has entered them, however it is not expected that those notes are saved between uses of the site.
- ✓ Follow best practices as outlined in lectures and exercises.
  - ○ Ensure JavaScript is used appropriately to provide a consistent and maintainable code across all relevant pages.
  - ○ Ensure your implementation still passes validation after modifications.

Notes:

Do not worry about making user interactions persistent (i.e. user changes remain after a refresh) at this stage.

Submitting your work:

You will be presenting your work with your group for review by a tutor during your tutorial session.

- • Be sure to have your work ready **at the start of your tutorial session** and **loaded on a laptop or USB Flash Drive** (if none of your group carries a laptop).
- • Submit your work in a ZIP file to the link on the course website before the end of your session.
- • Your tutor review your work and ask questions before assigning a mark.
- • Your mark will be available in the course website's gradebook before your next session unless otherwise noted.

Part 3 – Google Calendar & setting up your server

Before you begin:

- ✓ Review practical exercises 3-6
- ✓ Ensure NPM, NodeJS and Express are installed on the machine you are using (see details MyUni if unsure how to do this)
- ✓ Get a Google API Key

Task:

Further develop and implement your design.

- ✓ Follow best practices as outlined in lectures and exercises.
  - ○ Ensure modifications don't affect validation of your site.

Referring to the appropriate APIs and documentation you should now be able to begin to integrate Google Calendar Events into your website on the client side.

- ✓ Add the ability for users to make entries and notes about their day based around the Calendar Events from that day.
  - ○ Users should be able to see items from a specific day, rather than all items, even if users cannot yet choose or distinguish between days.
  - ○ Users should be able make notes on/around items even if those notes are not saved between uses of the site.

It's time to set up the server so your website can be accessed on other computers!

- ✓ Following the lecture notes and documentation on MyUni, set up an Express server and migrate your website to it.
- ✓ Test to ensure all parts of your website are working as expected now that they are being served by the Express server.

Now that your server is working, you'll need to start thinking about the content that your web application will be dealing with, and where it belongs.

- ✓ List the different pieces of content/information that your web application will be dealing with & determine where it should be stored, what format, and where it should be processed.
  - ○ Where does the information come from?
  - ○ What form should it take?
  - ○ If the information is on the server, what will the client need to send to retrieve that data?
  - ○ If the information is on the client, how will it be sent to the server?
  - ○ Does all information need to be stored on the server?
  - ○ What processing needs to be done to make the data useful?

Notes:

Do not worry about making user interactions persistent (user changes remain after a refresh) yet – that's coming next.

Submitting your work:

You will be presenting your work with your group for review by a tutor during your tutorial session.

- Be sure to have your work ready **at the start of your tutorial session** and **loaded on a laptop or USB Flash Drive** (if none of your group carries a laptop).
- Submit your work in a ZIP file to the link on the course website before the end of your session.
- Your tutor review your work and ask questions before assigning a mark.
- Your mark will be available in the course website's gradebook before your next session unless otherwise noted.

Part 4 – Client-server integration

Before you begin:

- ✓ Review practical exercises 5-8

Task:

You're now ready to start handling data on the server.

- ✓ Using a combination of GET/POST methods and AJAX, modify your website and server to implement the calls needed to handle the content/information identified in your list from Part 3.
- ✓ Use variables on your webserver to store content provided by the client.
  - ○ The data should remain between browser refreshes, however will be lost when the server is restarted.

Referring to the appropriate server APIs and documentation, improve your login system by integrate OpenID into your website on the server side.

- ✓ Integrate OpenID using Google as the identity provider into your web application's login system.

The main functionality of your web application should now be almost complete.

- ✓ Continue to build your web application to allow users to make entries and notes about their day based around the Calendar Events and Tasks from that day and view previous entries, even following a page refresh.

Notes:

The next part will cover integrating Databases into your web application to store the data currently being lost when the server closes.

Submitting your work:

You will be presenting your work with your group for review by a tutor during your tutorial session.

- • Be sure to have your work ready **at the start of your tutorial session** and **loaded on a laptop or USB Flash Drive** (if none of your group carries a laptop).
- • Submit your work in a ZIP file to the link on the course website before the end of your session.
- • Your tutor review your work and ask questions before assigning a mark.
- • Your mark will be available in the course website's gradebook before your next session unless otherwise noted.

Part 5 – Adding database storage

Before you begin:

- ✓ Review practical exercises 8-10
- ✓ Ensure MariaDB is installed and set up on the machine you are using (see details MyUni if unsure how to do this)

Task:

It's time to figure out how to organise and manage the large amounts of data that your web application will have! Start by planning your database.

- ✓ Design and draw an E-R diagram for your web application.
  - ○ Ensure that you have identified appropriate keys and defined the cardinality of relationships.
- ✓ Convert your E-R diagram into a text SQL Schema.

Now that you've designed your database, you can set it up and begin writing queries.

- ✓ Setup your SQL database and tables.
- ✓ Write down the queries needed to store and retrieve data in your database.

Once your database has been built, and queries written, you can now integrate it with your web application.

- ✓ Integrate your database and web application.
- ✓ Follow best practices as outlined in lectures and exercises.
  - ○ Ensure SQL statements follow the recommended style.
  - ○ Ensure your implementation is safe against SQL injection.

Notes:

Your final presentation is next week! Take some time to review this document and ensure that your application meets this spec.
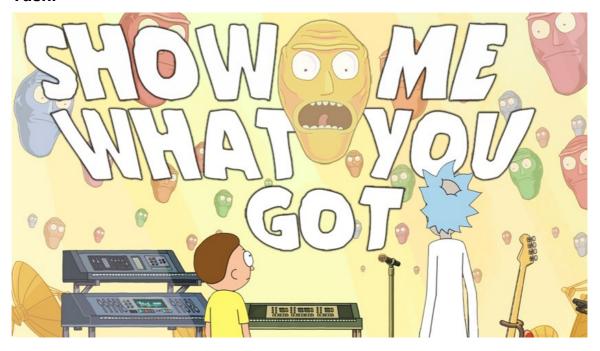
Submitting your work:

You will be presenting your work with your group for review by a tutor during your tutorial session.

- • Be sure to have your work ready **at the start of your tutorial session** and **loaded on a laptop or USB Flash Drive** (if none of your group carries a laptop).
- • Submit your work in a ZIP file to the link on the course website before the end of your session.
- • Your tutor review your work and ask questions before assigning a mark.
- • Your mark will be available in the course website's gradebook before your next session unless otherwise noted.

Final Presentation

Before you begin:

- ✓ Review this document to ensure you haven't missed anything

Task:

Notes:

Good luck!

Submitting your work:

The deadline for this assignment is the start of your scheduled practical.

Submit your work in a ZIP file to the link on the course website **before** the start of your practical session.

You will be presenting your work with your group for review by the teaching team during your scheduled Practical.

- Be sure to have your work ready and **loaded on a laptop or USB Flash Drive** (if none of your group carries a laptop).
- Submit your work in a ZIP file to the link on the course website before the **start** of the session.
- We will look at your work and ask questions before assigning a partial mark. Once presentations are complete we will review your online submission and finalise the mark.
- Your final mark will be available in the course website's gradebook within 2 weeks of your session unless otherwise noted.