Osmose

What is it:
A platform to facilitate interaction between students and instructors in a course.

Why Osmose:
os·mose /ˈäzmōsˈäs-/: The process of gradual or unconscious assimilation of ideas and knowledge.

The Problem:
Currently it is tough for students to know their peers and instructors within a module. The platform aims to allow students to get to know their peers better and encourages the spirit of sharing knowledge and learning together.

Students actually **do** like to help one another, as can be seen from the forums. However, the IVLE forum is ugly, messy and does not make people feel like using it.

Osmose will be a platform to facilitate the interaction and sharing between students and instructors of a course. Students can ask questions, answer questions from their peers, and also share new findings with one another.

The platform also aims gamify the user experience with a points and achievements systems Students and instructors can gain points when their posts (questions, answers and comments) are upvoted by their peers and earn achievements as a proof of recognition of their expertise.

Execution Plans:

Tech Stack	Roles and Responsibilities
● Front end: AngularJS, Foundation 4 ● Back end MVC: Sails.js ● Web Server: Node.js ● Hosting: EC2	● Soedar ○ Engineering lead ● Viet Tien ○ Back end ○ Server in-charge

	• Chun Mun
	  ○ Middleware
	• Yang Shun
	  ○ Product lead
	  ○ Designer
	  ○ Front end

Aspiration 1 - Application Type:

Choose to do an iframe application or a standalone application or both. Choose wisely and justify your choice to us with a short write-up.

We choose standalone. The downside of iframe apps is that there is less real-estate. Osmose also needs more space to display discussion threads and iframes severely limit the available height of the app. Also, Osmose is a platform that focuses on user content and having the Facebook frame is distracting.

In future, we may extend Osmose to allow users to sign up/log in via other services such as Twitter and Google+. Building a standalone app will give such flexibility.

Aspiration 2 - Application Name:

Your new baby needs a name! Give it one!

The name of the application is Osmose.

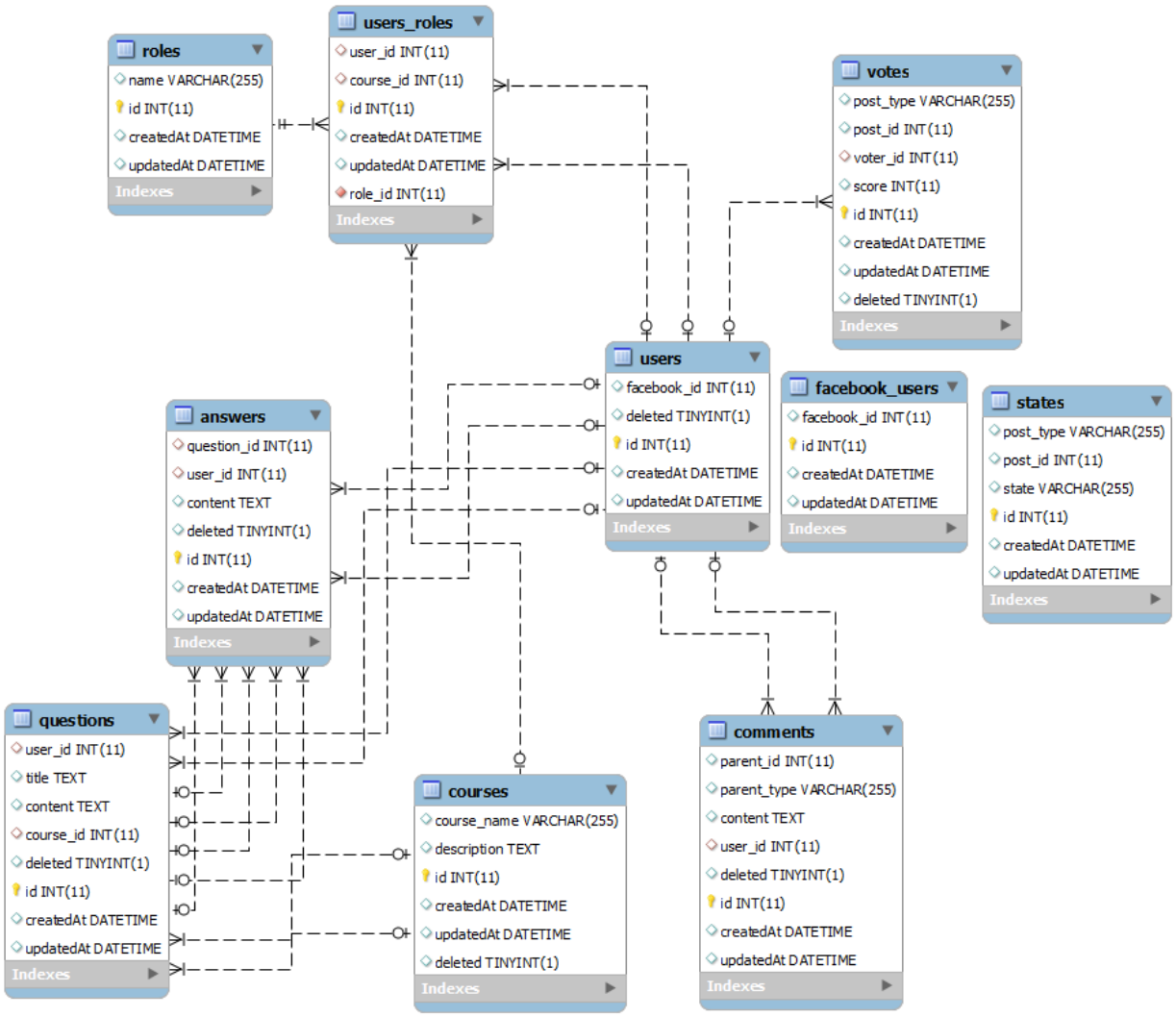
Aspiration 3 - Application Icon:

Aspiration 4 - Standalone or iframe

Integrate your application with Facebook. If you are developing an IFrame app, then users should be able to visit your app and at least see their name (retrieved using the API) on the page. Similarly, if you are developing a standalone app, users should be able to login to your app using their Facebook account and see their own name appearing.

You can login via Facebook at **http://osmose.soedar.com**.

Aspiration 5 - Database Schema:



Answers to Bonus Aspirations:

What are the pros and cons of each method of visibility control? When should one use the Javascript method and when should one use the PHP method?

This is an issue of client side (JS) vs server side (PHP) rendering.

Client side

Pros	Cons
● Can toggle between two states without 'refreshing' the page. We can do an ajax poll in the background to see if the user has already signed in to Facebook and update the state of the button if there is a change in the state.	● Server response has to include HTML of both buttons when only one has to be used. ● The javascript code to hide the irrelevant button is an unimportant part of the client side logic. ● Users may see the initial state of the page where both buttons are shown. ● Client has to query the server for the state of the user (logged in or not). ● If user disabled Javascript in his browser, both buttons will appear, and it could be a confusing experience for the user.

Server side

Pros	Cons
● Smaller server response size as only the HTML of one button has to be sent across.	● Server needs to know the state of the user (logged in or not).

When the visibility of a button should be toggled without refreshing the page, then a client side implementation would be better. However in this case, the state of the Logout button persists throughout the user's session and clicking Logout will bring the user to a new page that doesn't show logged in content, hence a server side implementation will be more suitable

What is the primary key of the home faculties table?

The primary key is the combination (matric_no, faculty)

Aspiration 6 - SQL Queries

Tell us some user queries (at least 3) in your app that needs database access. Provide the actual SQL queries you use and explain how it works.

Query	SQL Query	Explanation
Get information about a user.	SELECT * FROM users WHERE id = <user id>	We want to select all the user's information from the users table, if the id of the row matches the <user id>.
Get questions by a user and the question's comments.	SELECT * FROM questions q LEFT JOIN comments c ON q.id = c.parent_id WHERE q.user_id = <user_id> AND (c.parent_type IS NULL OR c.parent_type = 'QUESTION');	We want to get all the the question information and its related comments. During a left join, questions that have no comments will have null in its comments column, so we have to check for that.
Add a new comment	INSERT INTO comments (parent_id, parent_type, content, user_id) VALUES (<question id>, 'QUESTION', 'This is a comment, <user id>)	We want to insert a new comment by <user id> on a question with <question id>

Aspiration 7 - Facebook Graph Queries

Show us some of your most interesting Graph or FQL queries. Explain what they are used for. (2-3 examples)

Query	Facebook Graph	Explanation
Get information about a user	/me	Since the user will login from Facebook, we want some personal information from the user so that we would be able to identify them when they are on the application.
Get the user's friends	/me/friends?fields=installed,name, username	We want a way for the user to invite his/her friends to use our application. However, we do not want to get more information than necessary, so we only grab their username, full name, and whether they are already users of our application.
Get profile picture of a user	/<username>/picture	Grab the url of the user's profile picture, so we can make the user interface look pretty.

Aspiration 8 - Facebook Feed

We want some feeds! BUT remember to put thought into this. Nuisance feeds will not only earn you no credit but may incur penalization!

Questions can be easily shared on Facebook via on link by the bottom-right of every question. The user have complete control over the interesting stuff he wants to share.

Aspiration 9 - Facebook Like

Your application should include the Like button for your users to click on. Convince us why you think that is the best place you should place the button.

We have included the like button in two places, the landing page and the sidebar. We placed it at these places because:

Landing Page
When the user first land on the sweet landing page, they can like our application to encourage their friends to try out Osmose.

Sidebar
The sidebar exists on all the important functional pages of Osmose. As the user explores our application, and discover functionalities that they like, they can easily share it with their friends through the like button on the sidebar.

Aspiration 10 - Facebook Invite

First, include the feature to let users invite their friends to use your application. Next, create a page showing either (a) all friends who are users of your application; or (b) all friends (or some sensible subset) who are not users of your application.

Access the page through the **'Invite Friends'** link in the sidebar: http://osmose.soedar.com/invite

Aspiration 11 - Facebook Privacy

Explain how you handle a user's data when she removes your application. Convince us that your approach is necessary and that it is the most logical. Do also include an explanation on whether you violate Facebook's terms and condition.

When a user removes our application, we will remove the row associated with the user on the Facebook_user table, which stores all the information that we obtained from Facebook.

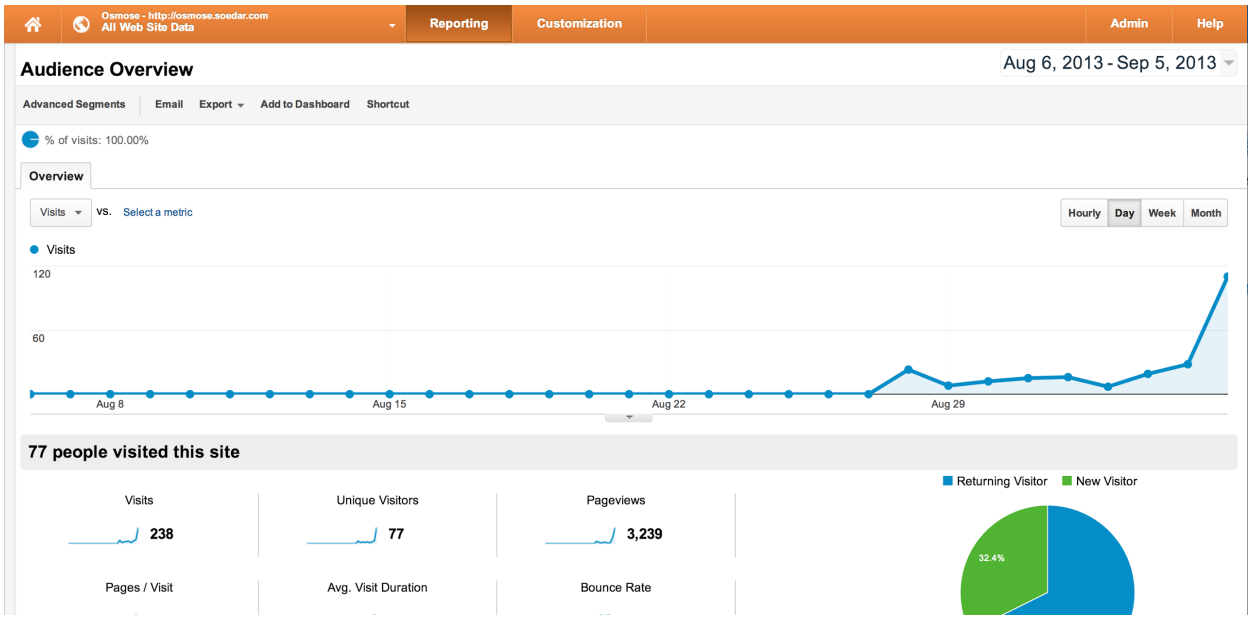
We have opted not to delete all the posts (questions, answers, comments) that the user have made. Doing so would break the flow of some threads, and would render them unintelligible. However, since we have

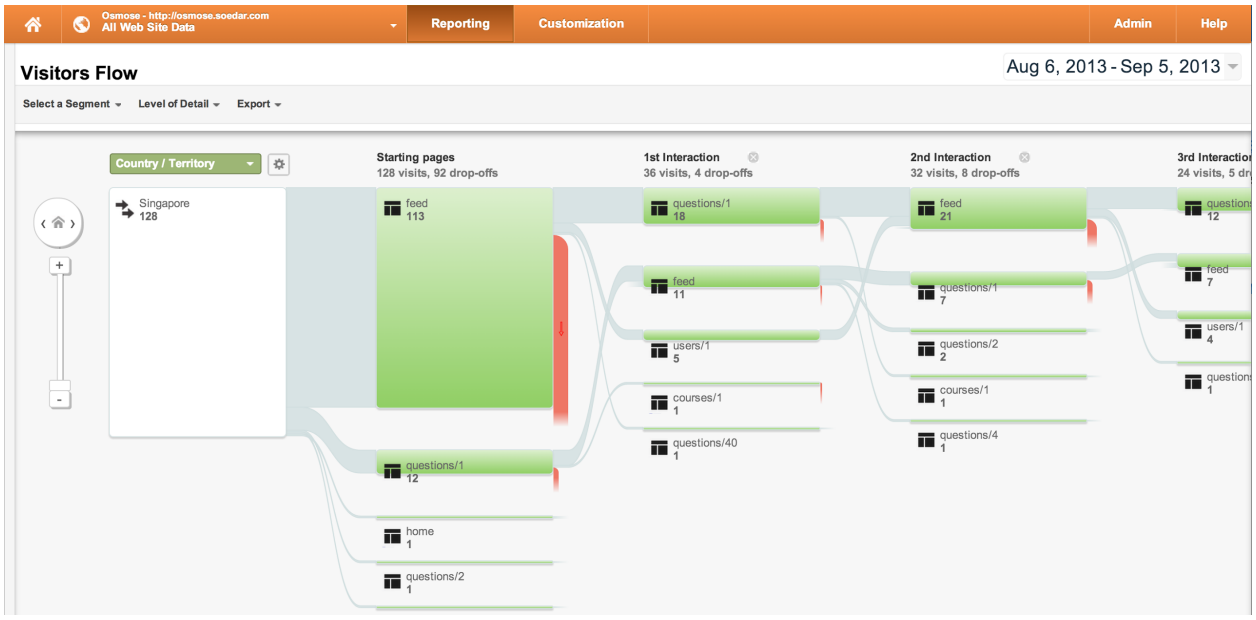
deleted all the personal information of the user, the posts made by the user will appear to have been made anonymously, and would be untraceable to the original poster on the server. Our team believes that this is a reasonable compromise that does not violate the Facebook's terms and condition.

We have not violated Facebook's terms and condition. It states that we have to delete all the user's data obtained from Facebook when the user request to remove our application. In addition, we have not collected additional data beyond the very basic that is needed to operate our application.

Aspiration 12 - Google Analytics

Embed Google Analytics on all your pages and give us a screenshot of the report. Note that this means you have to install Analytics at least 48 hours before submission deadline as Analytics only updates the report once per day.





Aspiration 13 - User Interactions

Describe 2-3 user interactions in your application and show us that you have thought through those interactions. It would be great if you could also describe other alternatives that you decided to discard, if any.

Feed as the Default View

After the user have logged in to Osmose, the user will be directed to the feed page. The feed page is a compilation of the recent activities on all the different courses that the user has subscribed to. (Right now, the feed page returns the recent activities sorted by time, but we can change it to return questions that we think the user would be interested) This allows the user to immediately see all the important and relevant questions immediately when they login to Osmose.

We have thought of putting the list of courses that the user have subscribed to as the main page once the user has logged in, but eventually did not choose to do that. When the users use Osmose, we think their purpose would be to check out the latest activity on their various courses. By putting the list of courses on the first page the users see, it means that the users would have to take an additional step to choose the course that they want before they can see it.

Post An Answer

On the question page, we have placed the answer entry box at the bottom of the page. One of the problems for a question-and-answer site is that there could be duplicated answers for the same question. By putting the answer entry box at the bottom, the user would hopefully have to go through most of the answers before deciding to add their answer for the question. This should prevent duplication, and would ensure that the quality of the answers remain high.

Upvote/Downvote of Answers
The users can choose to upvote or downvote the answers for the question to judge their quality. With a relatively large number of users, we can assume that a better answer would have a higher vote count. As we would want the user to quickly find the answers to the questions that they find are interesting, we will show the higher rated answers at the top, regardless of when the answer is posted.

Aspiration 14 - DOM Manipulation - AngularJS

Show us an interesting DOM manipulation through jQuery that occurs in your application.

jQuery is not applicable to us because in most pages of our application but we used a front end javascript framework called AngularJS. An interesting instance of DOM manipulation will be the search bar in the invite page. The thumbnails of friends shown are filtered according to the text input. AngularJS will only show friends with names that contain the string entered in the input box.

Aspiration 15 - Facebook Action and Object

Describe at least one Action and Object that you have created for your application and why you think it will create an engaging experience for the users of your app.

When an user posts a new question on Osmose, his facebook account will automatically create an Open Graph story with action “Ask” and object “Question”. The story will contain the title of the question as well as a brief description. The same happens when the user tries to answer a question, but with action “Answer” instead. Clicking on the story will bring users to that specific question on Osmose.

Because the default visibility of these stories is “Friends”, they will be likely to appear on users’ friends feeds. Therefore, Open Graph stories can help Osmose to attract more new users.

Aspiration 16 - Facebook Timeline

Describe how you have integrated with Timeline using Open Graph Collections.

Each user using Osmose will have two Open Graph Collections: the recent questions he asks and the recent questions he answers. These collections will appear in users’ timelines as part of their activities.

Aspiration 17 - Animations

Describe 2 to 3 uses of animations in your application. Explain what kind of value do they add to the context to which they are applied. Highlight your more innovative animations. We would be interested to know! (Optional)

Animation 1: Thumbnails of existing users fading in on the landing page. It improves the credibility of the platform that there are real users using it and gives the platform a sleek feel.

Animation 2: Loading of questions content and translating the content up and fading in after it loads.

Moving and appearing things brings the attention of the users to the part of the webpage that we want them to focus on: the content.

Aspiration 18 - AJAX/Socket.io

Describe any cool utilization of AJAX in your application.

All questions, answers and comments are subscribed to the publisher/subscriber system provided by Socket.io. On our server, any successful post from our users automatically cause it to emit an event and sends the corresponding post to users viewing it. For example, when a user post an answer to a question, all the other users viewing the question will receive an event notifying them that a user have posted a new answer to the question.

On the client side, we employ AngularJS to respond to the event received from Socket.io, and would only update the relevant parts of the Document Object Model (DOM). Following the previous example, we would add new answers to the correct location, sorted by the number of votes.

The result is a snappier experience for the user, and the user would not have to reload to see new updates to the question.

Aspiration 19 - jQuery Plugins

Tell us how you have made use of any existing jQuery plugins to enhance your application functionality.

jQuery Thumbnails - To handle the appending of profile picture thumbnails to the main page.
jQuery Autosize - Expand textarea when it spans multiple lines of content. So that users will know what they type.