

Assignment 2: Technical Report

By Andrew Parkhill (s5064284), Luke Powell (s5066140),
Daniel Pye (s4922402) and Michael Whatley (s5008606)

Link and Credentials

Our system is accessible at: <http://student40144.bucomputing.uk/Assignment-2/user/login/>

As you can see, we were able to remove “index.php” from the URL which makes the URL more compact and slightly improves SEO.

As stated in Appendix 2, there are 7 users already on the system: 4 students, 2 lecturers and 1 administrator. User registration can also be used to add more users.

Credentials

I recommend logging in as the users in bold to see the most interesting information.

Role	Name	Email	Password
Student	Andrew Moore	sandrewmoore@uni.ac.uk	7v4y3052
Student	Courtney Marsh	scourtneymarsh@uni.ac.uk	qgbTyihbxo
Student	Dan Reed	sdanreed@uni.ac.uk	68o0xikxzs
Student	Luke Wilson	slukewilson@uni.ac.uk	2fcRdaQ
Lecturer	Amy Mathis	amathis@uni.ac.uk	7wv989YJ
Lecturer	Kevin West	kwest@uni.ac.uk	ird5ctSUeg
Administrator	Nick Frost	nfrost@uni.ac.uk	stbRDYf4

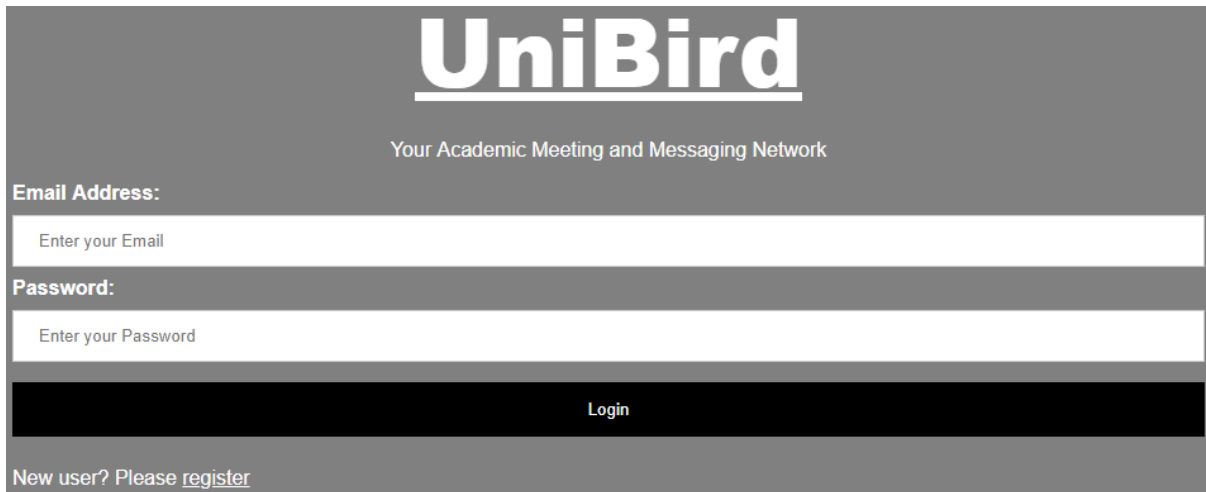
System Design Rationale

In considering the overall design of the application, it was agreed that we should aim for a web-based application style as opposed to the typical website layout. We agreed that the most professional layout would utilise a login page as the landing page which also allows us to ensure security from this point forward.

We also aimed for consistency throughout the application by implementing a navigation bar within the header which runs consistently throughout the application, adding and removing menu items dependent on the user's individual access rights using sessions.

Forms and tables are used to display and edit information with clear buttons included to ensure it is convenient for users to reset fields if necessary.

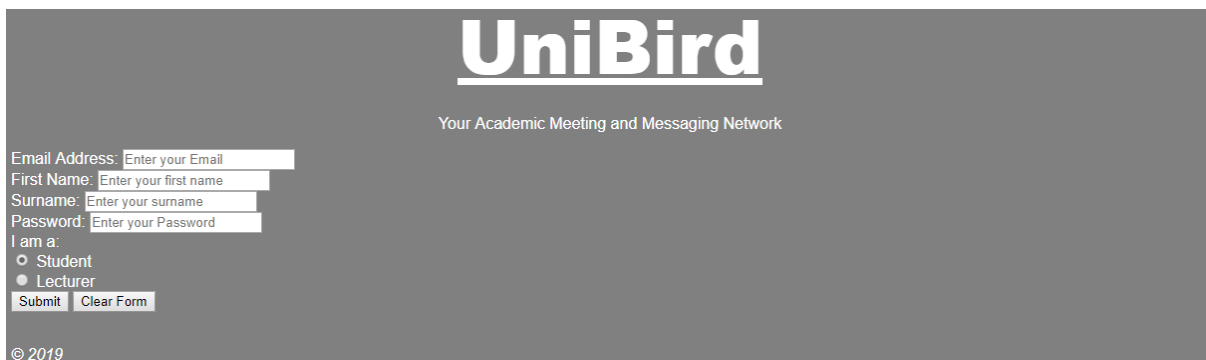
Login



The login form for UniBird features a dark grey header with the 'UniBird' logo in white, underlined, and the tagline 'Your Academic Meeting and Messaging Network' below it. The form itself is white and contains two input fields: 'Email Address:' and 'Password:', each with a placeholder 'Enter your Email' and 'Enter your Password' respectively. A black 'Login' button is positioned below the password field. At the bottom, a link 'New user? Please [register](#)' is provided.

This is a simple login page with required forms using the validation rules set with Codeigniter. It checks the input password against the hashed version in the database using password_verify and displays any input errors to the user.

Register



The register form for UniBird has a dark grey header with the 'UniBird' logo and tagline. The form is white and includes input fields for 'Email Address:', 'First Name:', 'Surname:', and 'Password:', each with a placeholder. Below these is a section 'I am a:' with radio buttons for 'Student' and 'Lecturer'. At the bottom of the form are 'Submit' and 'Clear Form' buttons. A copyright notice '© 2019' is located at the very bottom of the form area.


The register page uses required forms to create a new user in the user table. The radio button is used to set the user's authorisation level. Any errors in the form are displayed to the user after an attempted submission. (Note: for security reasons new users cannot register as an administrator)

Profile

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

Nick's Profile



[Edit Profile](#)

Name: Nick Frost
Role: Administrator
Email: nfrost@uni.ac.uk

Bio:
'Be yourself, everyone else is already taken.' -Oscar Wilde

© 2019

The profile page displays the user's data stored in the database and a picture based on their authorisation level as well as a link to the edit profile page. Bio and assigned lecturer information are protected by the user's profile visibility settings.

Edit Profile

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

Bio

'Be yourself, everyone else is already taken.' -Oscar Wilde

Profile Visibility

☐ All Users

☒ Lecturers Only

This page allows users to edit their bio and change the visibility setting for their profile. If set to lecturers only, their bio and assigned lecturer will be hidden from their profile to student users.

Messages

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

Announcements

You have no announcements!

Messages

[New Message](#)

From	Subject	Date		
sandrewmoore@uni.ac.uk	Forgot Password	2019-04-03 13:49:00	View	Delete
sdanreed@uni.ac.uk	No Assigned Lecturer	2019-04-03 15:29:26	View	Delete
kwest@uni.ac.uk	Re: Unassigned Student	2019-04-03 15:38:49	View	Delete

© 2019

The messages or inbox page shows the messages that have been sent to the user as well as announcements made by the student's assigned lecturer, or the lecturer's announcements if the user is a lecturer. The user can delete their messages here.

View Message

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

Forgot Password

From: sandrewmoore@uni.ac.uk

Hi Nick Frost, I am writing to let you know that I have forgotten my password for the university computer network, can it be reset please? Kind regards, Andrew Moore

© 2019

This page displays the details of a requested message.

New Message

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

New Message

To:

Subject:

Message Body:

Your Message

© 2019

The new message page allows users to send messages to other users, if accessed by clicking a message link on the address book page the 'To' field will contain the selected user's email address. The 'To' and 'Subject' forms are required, and any input errors are displayed to the user upon an attempted submission.

Meetings

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

Meetings

Browse your meetings with students

[New Meeting](#):

Meeting

Meeting with [Luke Wilson](#) - Support with Script Assignment: 2019-04-09 15:30:00 to 2019-04-09 16:00:00 in F201 [Delete Meeting](#)

© 2019

The meetings page displays any arranged meetings between a student and lecturer. Users can delete their meetings here.

Arrange Meeting

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[Logout](#)

Arrange Meeting

Select one of your assigned lecturer [Amy Mathis](#)'s available timeslots to arrange a meeting with them.

Timeslot

From 2019-04-04 14:00:00 to 2019-04-04 14:20:00 in P201 [Select Timeslot](#)

From 2019-04-04 14:20:00 to 2019-04-04 14:40:00 in P201 [Select Timeslot](#)

From 2019-04-05 11:30:00 to 2019-04-05 11:45:00 in B326 [Select Timeslot](#)

From 0000-00-00 00:00:00 to 0000-00-00 00:00:00 in ssssssss [Select Timeslot](#)

Meeting Title

This page displays the timeslots that have been created by the lecturer user or the student's assigned lecturer and allows them to select one and create a title for the meeting. If the user is a lecturer, the page will list their assigned students and ask them to select one. Once a time slot has been selected it is no longer available to select on this page.

Address Book

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

Address Book

Browse all users:

Role	Name	
Student	Moore, Andrew	Message
Lecturer	Mathis, Amy	Message
Administrator	Frost, Nick	Message
Student	Marsh, Courtney	Message
Student	Wilson, Luke	Message
Lecturer	West, Kevin	
Student	Reed, Dan	Message Add Student

© 2019

This page lists all registered users with links to their profiles as well as a link to the create message page which will fill the form with the selected user's registered email. For lecturers an add student link is shown for any students without an assigned lecturer which will add the

selected student to the lecturer's assigned students. For administrators a delete user link is also shown.

My Students

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

My Students

Browse students assigned to you:

Name	
Wilson, Luke	Remove Student

© 2019

The 'my students' page is only accessible to lecturers and administrators; it displays a list of their assigned students as well as a link to remove students from their list.

My Announcements

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

My Announcements

[New Announcement](#)

Subject	Date		
Script Assignment Upload	2019-04-03 15:01:59	View	Delete
Script Assignment Support	2019-04-03 15:05:20	View	Delete

© 2019

This page is only accessible to lecturers and administrators, it shows a list of announcements made by that user. Announcements can be deleted here.

View Announcement

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

Relational Databases Revision

From: amathis@uni.ac.uk

Hi Students, I have sent out the practice questions for the upcoming Relational Databases exam by email. Any queries feel free to message or arrange a meeting. I have set up some timeslots for tomorrow and the day after. Best, Amy

© 2019

This page displays the details of a requested announcement.

New Announcements

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

New Announcement

Subject:

Announcement:

© 2019

This page is only accessible to lecturers and administrators, it allows a lecturer to send an announcement to their assigned students. Both form fields are required and show any input errors to the user on a submit.

My Timeslots

UniBird

[My Profile](#)[Messages](#)[Meetings](#)[Address Book](#)[My Students](#)[My Announcements](#)[My Timeslots](#)[Logout](#)

My Timeslots

Browse your timeslots:

Timeslot	
From 2019-04-09 09:30:00 to 2019-04-09 10:00:00 in F201	Delete Timeslot
From 2019-04-10 12:05:00 to 2019-04-10 12:20:00 in F201	Delete Timeslot
From 2019-04-10 12:25:00 to 2019-04-10 12:40:00 in F201	Delete Timeslot

Add Timeslot

Start Date/Time

dd/mm/yyyy --:--

End Date/Time

dd/mm/yyyy --:--

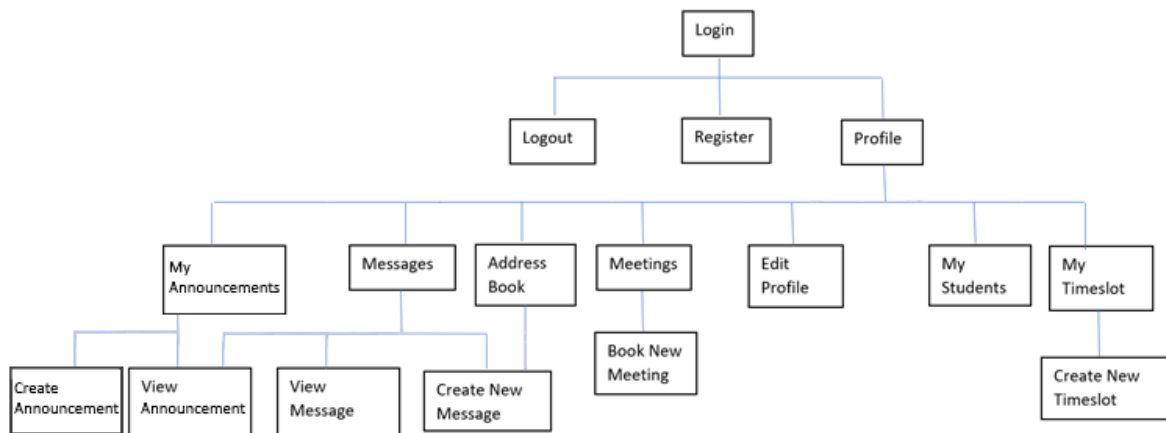
Location

© 2019

This page is only accessible to lecturers and administrators, it allows lecturers to specify a time, date and location for a meeting to take place. Once a time slot has been created it will be displayed on the 'Arrange meeting' page for users to select. Created timeslots can be deleted here.

Navigation Design

Detailed below is a navigation structure for the site:



Database Design Rationale

Our system is driven by a MySQL database which contains the following tables:

User

Information about all users of the system.

- **usr_id** system-friendly unique identifier, auto-increments to ensure uniqueness.
- **usr_email** user-friendly naturally unique identifier. Unlike `usr_id`, we can display and expect users to remember this.
- **usr_my_key** stores the user's password for authentication, hashed to improve security. This column is awkwardly named as an added security measure against unauthorised database access.
- **usr_first_name** stores the user's forename for displaying.
- **usr_last_name** stores the user's surname for displaying.
- **usr_bio** stores user-defined self-descriptive text, nullable.
- **usr_assigned_lecturer_id*** points to the user's assigned lecturer if they are a student, nullable.
- **usr_auth_level** stores the user's role: 0 = student, 1 = lecturer, 2 = administrator, enables extra functionality for lecturers and administrators, default is 0 to improve security.
- **usr_profile_is_private** stores whether or not students can view protected parts of the user's profile, satisfies personal security needs, default is false to improve data availability.

Message

Information about private messages sent using the system.

- **msg_id** system-friendly unique identifier, auto-increments to ensure uniqueness.
- **msg_author** stores the author's email address for clarification display purposes.
- **msg_subject** stores the headline of the message for displaying.
- **msg_recipient** stores the recipient's email address for clarification display purposes.
- **msg_date** stores the non-localised server-side date and time that the message was sent, for displaying.

- **msg_body** stores the content of the message for displaying.
- **msg_read** stores whether or not the message has been read by its recipient, default is false to align with logic.

Timeslot

Information about times and places that lecturers have declared as available for meeting with their assigned students.

tsl_id system-friendly unique identifier, auto-increments to ensure uniqueness.

tsl_start stores the non-localised server-side date and time that the timeslot begins, for displaying.

tsl_end stores the non-localised server-side date and time that the timeslot ends, for displaying.

tsl_lecturer_id* points to the lecturer that owns this timeslot.

tsl_booked stores whether or not a meeting has been arranged using this timeslot, default is 0 to align with logic.

tsl_location stores the location where the timeslot takes place, for displaying.

Meeting

Information about meetings between students and their assigned lecturers.

- **met_id** system-friendly unique identifier, auto-increments to ensure uniqueness.
- **met_time_slot_id*** points to the time slot used for this meeting.
- **met_title** stores the headline for the meeting, for displaying.
- **met_lecturer_id*** points to the meeting's lecturer attendee.
- **met_student_id*** points to the meeting's student attendee.

Security Considerations

As a development team we took the decision to implement password hashing, which ensures a level of security from the outset. The password is hashed at registration stage, which ensures an encrypted version is held in the database and later used for validation.

Sessions have been implemented to further secure access rights. The authorisation level is assigned to a session array at login stage. This is then utilised throughout the application to ensure that the relevant information is displayed to the user i.e. ensuring lecturers and students only have access to the information they require

MVC Framework/ Version Control

As a group we made use of the CodeIgniter MVC framework which facilitated the separation of code in to a logical structure and allowed for a more logical approach for the group to follow. The separation of code between Models, Views and Controls allowed for the separation of concerns and allowed all group members to take responsibility for areas of the development and resulted in more concise files which are more readable for larger web applications.

The group also made use of GitHub for Version Control purposes which worked very successfully, allowing all members to work remotely and notify when any changes had been pushed to the repository.

File Structure

Interesting items in **bold**, Directories in black, [Files in blue](#)

Assignment-2 (top-level directory, contains everything)
application (contains the entire MVC architecture)
 cache
 config (contains configuration information the system relies on)
 [autoload.php](#) (defines systems that are loaded by default)
 [config.php](#) (defines configuration information such as base URL)
 constants.php
 [database.php](#) (defines associated database credentials)
 doctypes.php
 foreign_chars.php
 hooks.php
 index.html
 memcached.php
 migration.php
 mimes.php
 profiler.php
 [routes.php](#) (defines custom URLs)
 smileys.php
 user_agents.php
 controllers (contains controllers that are part of the MVC architecture)
 [Meeting.php](#) (coordinates meeting functionality)
 [Message.php](#) (coordinates messaging and announcement functionality)
 [Timeslot.php](#) (coordinates timeslot functionality)
 [User.php](#) (coordinates user-related functionality)
 index.html
 core, helpers, hooks, language, libraries, logs
 models (contains models that are part of the MVC architecture)
 [Meeting_model.php](#) (manages the "meeting" database table)
 [Message_model.php](#) (manages the "message" database table)
 [Timeslot_model.php](#) (manages the "timeslot" database table)
 [User_model.php](#) (manages the "user" database table)
 index.html
 third_party
 views (contains views that are part of the MVC architecture)
 errors
 meeting (contains code to display meeting information)
 [arrangemeeting.php](#) (displays a create meeting form)
 [meetingsfooter.php](#) (displays the end of the meeting list)
 [meetingsheader.php](#) (displays the start of the meeting list)
 [meetingsitem.php](#) (displays a single meeting list item)
 [studentsselection.php](#) (displays a list of assigned students)
 [timeslotsselection.php](#) (displays a list of available timeslots)
 message (contains code to display message/announcement information)
 [createannouncement.php](#) (displays create announcement form)
 [inbox.php](#) (displays a list of messages)
 [messagesent.php](#) (displays a message sent confirmation)
 [myannouncements.php](#) (displays a list of announcements)
 [newmessage.php](#) (displays a create message form)
 [viewannouncement.php](#) (displays a single announcement)
 [viewmessage.php](#) (displays a single message)
 templates (contains code to display information across the whole system)
 [footer.php](#) (displays a standard copyright footer)
 [header.php](#) (displays standard header and navigation bar)
 timeslot (contains code to display timeslot information)
 [addtimeslot.php](#) (displays a timeslot creation form)
 [mytimeslots.php](#) (displays a list of timeslots)
 user (contains code to display user-related information)
 [addressbook.php](#) (displays a list of users)
 [editprofile.php](#) (displays a profile editing form)
 [login.php](#) (displays a login form)
 [mystudents.php](#) (displays a list of assigned students)
 [profile.php](#) (displays information about an individual)
 [register.php](#) (displays a registration form)
 [registersuccess.php](#) (displays registration confirmation message)
 index.html
 .htaccess
 index.html
 css (contains Cascading Style Sheets (CSS) styling information)
 [style.css](#) (contains styling information applied to the entire system)
 images (contains image assets)
 [administrator_pic.png](#) (default profile picture for administrator users)
 [lecturer_pic.png](#) (default profile picture for lecturer users)
 [student_pic.png](#) (default profile picture for student users)
 sql_scripts (contains Structured Query Language (SQL) scripts for reference only)
 [create_all](#) (creates all the tables)
 [create_meeting.sql](#) (creates the meeting table)
 [create_message.sql](#) (creates the message table)
 [create_timeslot.sql](#) (creates the timeslot table)
 [create_user.sql](#) (creates the user table)
 [drop_all.sql](#) (drops all the tables)
 [insert_all.sql](#) (inserts sample data into all tables)
 [insert_meeting.sql](#) (inserts sample data into the meeting table)
 [insert_message.sql](#) (inserts sample data into the message table)
 [insert_timeslot.sql](#) (inserts sample data into the timeslot table)
 [insert_user.sql](#) (inserts sample data into the user table)
 system (contains core CodeIgniter libraries and helpers etc.)
 user_guide (contains CodeIgniter documentation)
 .editorconfig
 .gitignore
 [README.md](#) (description of the project)
 [blank_config.php](#) (template copy of config.php without environment-specific details)
 [blank_database.php](#) (template copy of database.php without environment-specific details)
 composer.json
 contributing.md
 Index.php (takes controllers and views as parameters to display the system)
 license.txt
 readme.rst

[1468 words]

References

- W3Adda. No Date. Codeigniter Directory Structure | W3Schools | Tutorialspoint | W3Adda. [ONLINE] Available at: <https://www.w3adda.com/codeigniter-tutorial/codeigniter-directory-structure>. [Accessed 2 April 2019]

Appendix 1: System Testing

Test Cases

ID	Purpose	Test Procedure	Expected Outcome	Actual Outcome	Pass/Fail
Registration					
1	Ensure that registration functions and sends details to database	Enter all required fields in registration page	All details sent to database including a hashed password. Page redirects to profile page on successful registration	All details sent to database including a hashed password. Page redirects to profile page on successful registration	Pass
2	Email validation	Enter invalid email	Error message displays	Error message displays	Pass
3	Email database validation	Enter already registered email address at registration stage	Error message displays	Database error displays. Should implement a more specific error message	Fail
4	Checking required fields	Leave a field empty and try to submit form	Error message displays	Form resets without error message	Fail
4	Re-Test Checking required fields	Leave a field empty and try to submit form	Error message displays	Error message displays	Pass
User Authentication					
5	Ensure that login operates with correct credentials	Enter correct login credentials	Profile page for the user displays	Profile page for the user displays	Pass
6	Ensure that error message displays with incorrect credentials	Enter incorrect login credentials	Error message displays	Error message displays	Pass
7	Ensure that error message displays when required field not completed	Leave field empty and submit form	Error message displays	Page refreshes without error message	Fail
7	Re-test Ensure that error message displays when required field not completed	Leave field empty and submit form	Error message displays	Error message displays	Pass
Profile					

8	Check all the correct user credentials have been displayed	Ensure the loaded credentials match the correct record from the user table in the database	All the details loaded on the profile page match those of the logged in user	All the details loaded on the profile page match those of the logged in user	Pass
9	Check that correct links have been displayed in the navbar based on user's authentication level	Review navbar	My Timeslots, My Announcements and My Students only visible for lecturers	My Timeslots, My Announcements and My Students only visible for lecturers	Pass
10	Ensure the 'My Profile' link loads the correct profile	Click 'My Profile' link	Link reloads the same profile	Link reloads the same profile	Pass
11	Ensure edit profile details save and implement as expected	Change profile Bio and set visibility as Lecturer only	Bio updates in profile and database and bio not visible to students	Bio updates in profile and database and bio not visible to students	Pass
12	Check the "clear" button in the edit profile page clears the bio field	Click the clear button when text has been added to the bio	The bio field clears	Only the additional added text clears from the bio	Fail?
13	Check that switching visibility to all users functions	Change visibility to "all users"	Bio visible in student profile	Bio visible in student profile	Pass
Messages					
14	Check that all messages for the user have been displayed in the messages table	Review data	All details from the message table display for the logged in user	All details from the message table display for the logged in user	Pass
15	Check that composing and sending a new message operates successfully	Click 'New Message' Complete all fields Click Submit	Message displays to confirm that details have been successfully sent, and message table update with the new message	Message displays to confirm that details have been successfully sent, and message table update with the new message	Pass
16	Check that a composed message does not submit without required fields	Click new message and leave some fields empty before submitting	Error message displays	Form resets without error message	Fail
16	Re-test Check that a composed message does not submit without required fields	Click new message and leave some fields empty before submitting	Error message displays	Error message displays	Pass
17	Check how form responds to incorrect	Enter incorrect email credentials and submit form	Error message displays confirming user not found	Error message displays	Pass

	email address being entered			confirming user not found	
18	Check that clear button works on new message	Click clear button with text in fields	Text clears from form	Text clears from form	Pass
19	Check that delete message, deletes message successfully	Click delete button beside message	Message deletes from page and database	Message deletes from page and database	Pass
20	Check that view message functions	Click "view" beside required message	Message details display on separate page	Message details display on separate page	Pass
Announcements					
21	Check that announcement displays on student message page	Make announcement from lecturer profile	Announcement displays on student message page	Announcement displays on student message page	Pass
22	Check that delete announcement removes announcement from lecturer and student	Delete announcement while logged in as lecturer	Announcement removes from lecturer and student	Announcement removes from lecturer and student	Pass
23	Check that view announcement link functions	Click "view" link beside an announcement	All announcement details display	All announcement details display	Pass
24	Check that announcement form will not submit with empty fields	Click submit with empty fields	Error message should display	Form resets without error	Fail
24	Re-test Check that announcement form will not submit with empty fields	Click submit with empty fields	Error message displays	Error message displays	Pass
Address Book					
25	Check that search function searches for the details provided	Type a name in the search bar and click search	Returns all records containing that entered details	Returns all records containing that entered details	Pass
26	Check that search function only displays listed users	Type a name not listed in the Database in the search bar	No records should be returned	No records should be returned	Pass
27	Test message link beside a user in the address book	Click message link	Link takes you to new message form	Link takes you to new message form	Pass

28	Check that student cannot be added if they already have an assigned lecturer	Review address book list	Link to add student not available on those without assigned lecturer	Link to add student not available on those without assigned lecturer	Pass
29	When logged in as admin, ensure that delete user function operates	Click “delete” button beside user	User deleted from database	User deleted from database	Pass
Meetings					
30	Check that all scheduled meetings have been loaded from Database	Review data	All meetings involving the logged in user displayed	All meetings involving the logged in user displayed	Pass
31	Check the “new meeting” function	Click ‘New Meeting’ link	All scheduled time slots for the assigned lecturer displayed which allows the user to arrange a meeting	All scheduled time slots for the assigned lecturer displayed which allows the user to arrange a meeting	Pass
32	Check that form does not submit with empty fields	Leave fields empty and submit timeslot form	Error message should display	Form refreshes without error message	Fail
32	Re-test Check that form does not submit with empty fields	Leave fields empty and submit timeslot form	Error message displays	Error message displays	Pass
33	Check that user is unable to set two timeslots at the same time	Schedule two timeslots at the same time	Error message displays to state that timeslots cannot overlap	Form refreshes without displaying error message	Fail
34	Check ‘Select Timeslot’ link functions	Click ‘Select time slot’ on the assigned lecturers list of available slots	Meeting title text field should appear. On submission, user should be redirected to their updated list of meetings	Incorrect error message when final timeslot booked “Unfortunately, you cannot arrange a meeting because your assigned lecturer has no available timeslots.”	Fail
34	Re-test Check ‘Select Timeslot’ link functions	Click ‘Select time slot’ on the assigned lecturers list of available slots	Meeting title text field appears. On submission, user should be redirected to their updated list of meetings	Meeting title text field appears. On submission, user should be redirected to their updated list of meetings	Pass

35	Ensure meeting cannot be scheduled when lecturer does not have an assigned student	Click "select timeslot" link as a lecture with no assigned student	Error message displays confirming that meeting cannot be scheduled	Error message displays confirming that meeting cannot be scheduled	Pass
Logout					
36	Check logout function	Click logout in the navigation bar and try to navigate to a profile via URL	Page logs out, returns to login page and destroys session (unable to navigate to profile without logging in)	Page logs out, returns to login page and destroys session (unable to navigate to profile without logging in)	Pass

Appendix 2: Sample Data

We have compiled sample data for testing purposes.

Users

usr_id	usr_email	usr_my_key	usr_first_name	usr_last_name	usr_bio	usr_assigned_lecturer_id	usr_auth_level	usr_profile_is_private
1	sandrewmoore@uni.ac.uk	\$2v\$10\$79ro8hm3oA5VK4416RloulmuP17CNTotA6HaOEKvPmbvvruihO	Andrew	Moore	Hev!	2	0	1
2	amathis@uni.ac.uk	\$2v\$10\$A0T.u1w.JNvnf1ChYfwCxed26Yht.Rf9kVW8inbatDAFLcR74i	Amv	Mathis	Hello. I am the Programme Leader for Computin...	NULL	1	0
3	nfrost@uni.ac.uk	\$2v\$10\$9.WFOrhmvFVDELLlOKc56.bYwDc/aXUo/.8YHMF.bAOuFv14OU6	Nick	Frost	'Be yourself: everyone else is already taken.' ...	NULL	2	0
4	scourtnevmarsh@uni.ac.uk	\$2v\$10\$Kl95JDeD/VLPww.29Owww.IY540RAWWvEcJDb4hTT/O194ECCxO	Courtnev	Marsh	I am studino BSc (Hons) Software Enoinnering.	2	0	0
5	slukewilson@uni.ac.uk	\$2v\$10\$0e8ohooaDkzUomo3mOOx.colrPDaHdadd7BPBFdUO6koYoiU/AS	Luke	Wilson	Film student	6	0	1
6	kwest@uni.ac.uk	\$2v\$10\$F5SKZdSPXHTI96EH6TXkustI/iH0Fdaofo/TsYbs4u8hKJMIUAO	Kevin	West	Film Production and Cinematooarshv lecturer	NULL	1	0
7	sdanreed@uni.ac.uk	\$2v\$10\$XPo6HesDTooM6l8nNShaeGBV1o5Z6VPHPhrJcVMMVY.xoSMEddMa	Dan	Reed	NULL	NULL	0	0
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

The “user” table has been populated with 7 users: 4 students, 2 lecturers and 1 administrator. A student can manage their profile, messages and meetings. A lecturer can additionally manage their student list, timeslots and announcements (to their assigned students). An administrator can additionally delete users. As you can see, our passwords are hashed to ensure security (i.e. the database never actually stores the real password for any user). Furthermore, the name of this field is deliberately unintuitive as a small added security measure against unauthorised database access. In the future we could enhance our protection of user data by hashing additional fields, especially “usr_email” because this information is also used to log into the system.

Messages

msg_id	msg_author	msg_subject	msg_recipient	msg_date	msg_body	msg_read
1	sandrewmoore@uni.ac.uk	Foroot Password	nfrost@uni.ac.uk	2019-04-03 13:49:00	Hi Nick Frost. I am writing to let you know that I...	0
2	nfrost@uni.ac.uk	Re: Foroot Password	sandrewmoore@uni.ac.uk	2019-04-03 13:59:29	No problem Andrew. it has been reset to 'i188K...	0
3	amathis@uni.ac.uk	Relational Databases Revision	announcement	2019-04-03 14:17:39	Hi Students. I have sent out the practice questi...	0
4	scourtnevmarsh@uni.ac.uk	Relational Databases O3	sandrewmoore@uni.ac.uk	2019-04-03 14:23:49	hev andv. have you done question 3 for relatio...	0
5	sandrewmoore@uni.ac.uk	Re: Relational Databases O3	scourtnevmarsh@uni.ac.uk	2019-04-03 14:26:27	No I don't think we covered unions did we? I'll a...	0
6	sandrewmoore@uni.ac.uk	Unions	amathis@uni.ac.uk	2019-04-03 14:32:20	Hi Amv. I see that question 3 of the relational d...	0
7	amathis@uni.ac.uk	Re: Unions	sandrewmoore@uni.ac.uk	2019-04-03 14:36:09	Hi Andrew. ves we covered unions duringa lectur...	0
8	scourtnevmarsh@uni.ac.uk	Re: Re: Relational Databases O3	sandrewmoore@uni.ac.uk	2019-04-03 14:39:00	Yeah ikr? Anvwav I arranged to meet Amv on F...	0
9	slukewilson@uni.ac.uk	Script Assianment	kwest@uni.ac.uk	2019-04-03 14:46:00	Hi Kevin. I am wonderina what is the preferred ...	0
10	sandrewmoore@uni.ac.uk	Unions Lecture	scourtnevmarsh@uni.ac.uk	2019-04-03 14:49:46	So she said we covered unions in lecture 6. we ...	0
11	kwest@uni.ac.uk	Re: Script Assianment	slukewilson@uni.ac.uk	2019-04-03 14:56:58	Hi Luke. thanks for your messaoe. I would pref...	0
12	kwest@uni.ac.uk	Script Assianment Upload	announcement	2019-04-03 15:01:59	Hi. iust writing to let you know that the assianm...	0
13	kwest@uni.ac.uk	Script Assianment Support	announcement	2019-04-03 15:05:20	Hi All. I foroot to mention that I also set up som...	0
14	amathis@uni.ac.uk	Mid-Unit Proaress Review	announcement	2019-04-03 15:14:54	Hi class. I have arranged Mid-Unit Proaress Rev...	0
15	scourtnevmarsh@uni.ac.uk	Mid-Unit Proaress Review	amathis@uni.ac.uk	2019-04-03 15:18:53	Hi Amv. I am writino to let you know that I will b...	0
16	amathis@uni.ac.uk	Re: Mid-Unit Proaress Review	scourtnevmarsh@uni.ac.uk	2019-04-03 15:23:49	Hi Courtnev. thank you for lettina me know. Th...	0
17	scourtnevmarsh@uni.ac.uk	Re: Re: Mid-Unit Proaress Review	amathis@uni.ac.uk	2019-04-03 15:25:43	Thank you. I look forward to meeting with you t...	0
18	sdanreed@uni.ac.uk	No Assianed Lecturer	nfrost@uni.ac.uk	2019-04-03 15:29:26	Hi. mv friend Luke said that our lecturer Kevin s...	0
19	nfrost@uni.ac.uk	Re: No Assianed Lecturer	sdanreed@uni.ac.uk	2019-04-03 15:33:24	Hi Dan. sorrv to hear that you are experiencina...	0
20	nfrost@uni.ac.uk	Unassianed Student	kwest@uni.ac.uk	2019-04-03 15:36:29	Hi Kevin. a student. Dan Reed has let me know ...	0
21	kwest@uni.ac.uk	Re: Unassianed Student	nfrost@uni.ac.uk	2019-04-03 15:38:49	Hi Nick. thank you for lettina me know. Yes Dan ...	0
NULL	NULL	NULL	NULL	NULL	NULL	NULL

The "message" table has been populated with a combined total of 21 private messages between all the users and announcements from lecturers to their assigned students.

Timeslots

tsl_id	tsl_start	tsl_end	tsl_lecturer_id	tsl_booked	tsl_location
1	2019-04-04 10:45:00	2019-04-04 11:00:00	2	0	P203
2	2019-04-04 14:00:00	2019-04-04 14:20:00	2	0	P201
3	2019-04-04 14:20:00	2019-04-04 14:40:00	2	0	P201
4	2019-04-05 11:30:00	2019-04-05 11:45:00	2	0	B326
5	2019-04-05 11:45:00	2019-04-05 12:00:00	2	1	B326
6	2019-04-09 09:30:00	2019-04-09 10:00:00	6	0	F201
7	2019-04-09 15:30:00	2019-04-09 16:00:00	6	1	F201
8	2019-04-10 12:05:00	2019-04-10 12:20:00	6	0	F201
9	2019-04-10 12:25:00	2019-04-10 12:40:00	6	0	F201
10	2019-04-11 10:00:00	2019-04-11 10:20:00	2	1	P203
11	2019-04-11 10:20:00	2019-04-11 10:40:00	2	1	P203
13	2019-04-12 10:20:00	2019-04-12 10:40:00	2	1	P201
NULL	NULL	NULL	NULL	NULL	NULL

The “timeslot” table has been populated with 12 timeslots created by lecturers. A timeslot is a “time and a place” that can be used for a meeting. We assume that a lecturer is always busy unless otherwise specified, so they can declare “timeslots” during which they are available for students to arrange a meeting during. Lecturers can also use their own timeslots to book meetings with their assigned students too.

Meetings

met_id	met_time_slot_id	met_title	met_lecturer_id	met_student_id
1	5	Relational Databases Support	2	4
2	10	Mid-Unit Progress Review	2	1
4	13	Mid-Unit Progress Review	2	4
5	7	Support with Script Assignment	6	5
NULL	NULL	NULL	NULL	NULL

The “meeting” table has been populated with 4 meetings that use the above timeslots: 2 created by students with their assigned lecturers and 2 created by lecturers with specified assigned students. Our system currently treats meetings as a strictly “student and lecturer” arrangement. Students can arrange meetings with only their assigned lecturer and lecturers can arrange meetings with only their assigned students. If our system were to expand, we might need to change our infrastructure so that the definition of a “meeting” is more flexible.

WP Assignment 2 – Group Self-Reflection Sheet

	Yes (completed)	Partially completed	No (not submitted)
Major requirements			
User registration	X		
User authentication	X		
Sending messages	X		
Personal information /profile pages	X		
Meeting arrangement	X		
Listing/viewing	X		
Searching	X		
System features			
MVC used	X		
Sessions used	X		
Input validation and sanitisation	X		
Secure encryption via encoding or hashing	X		
Additional features and extra functionality			X

List the third party libraries, frameworks, scripts, etc. that you used in your assignment:

CodeIgniter

As a group, what is the best part of your work?

The use of sessions to set pages dependant on authorisation levels

As a group, what is the weakest part of your work?

We could have focused more effort on the design phase in order to improve the usability of our final product

As a group, which part of the assignment you struggled most with?

Differing programming styles made it difficult to maintain consistency in the code

As a group, have you experienced any challenges or issues?

Generally, a good experience. It was difficult to keep everyone to the same level, particularly in learning the use of the CodeIgniter framework