Course Outline

Contents

- Course Details
- Course Aims
- Learning Outcomes
- Assumed Knowledge
- Teaching Strategies
- Teaching Rationale
- Student Conduct
- Assessment
- Course Schedule
- Resources for Students
- Course Evaluation and Development

Course Details

Course Code	COMP9311
Course Title	Database Systems
Convenor	Rachid Hamadi (/users/z2286838)
Admin	Rachid Hamadi (/users/z2286838)
Classes	**Lectures** : Wednesday 15:00-18:00 ChemScM17 Timetable for all classes (/COMP9311/17s2/timetable)
Consultations	Rachid Hamadi: Wednesday 14:00-15:00 in K17 Level2 203 Consultation Room
Units of Credit	6
Course Website	http://cse.unsw.edu.au/~cs9311/ (https://webcms3.cse.unsw.edu.au/COMP9311/17s2)
Handbook Entry	http://www.handbook.unsw.edu.au/postgraduate/courses/current/COMP9311.html (http://www.handbook.unsw.edu.au/postgraduate/courses/current/COMP9311.html)

Course Aims

This course aims to explore in depth: the practice of developing database applications; and the theory behind relational database management systems (RDBMSs). It will also give an overview of the technologies used in implementing database management systems and the past, present and future of database systems and database research.

Large data resources are critical to the functioning of just about every significant modern computer application. Hence, knowledge of how to manage them is clearly important to the IT industry. In the context of further study, understanding how to use databases effectively is essential for courses such as COMP9321 (http://www.handbook.unsw.edu.au/postgraduate/courses/2017/COMP9321.html) Web Applications Engineering and COMP9322 (http://www.handbook.unsw.edu.au/postgraduate/courses/2017/COMP9322.html) Service-Oriented Architectures. COMP9311 also provides a foundation for further study in advanced database topics, such as COMP9315 (http://www.handbook.unsw.edu.au/postgraduate/courses/2017/COMP9315.html) Database Systems Implementation and COMP9318 (http://www.handbook.unsw.edu.au/postgraduate/courses/2017/COMP9318.html) Data Warehousing and Data Mining. Database concepts are also relevant in courses such as COMP9319 (http://www.handbook.unsw.edu.au/postgraduate/courses/2017/COMP9319.html) Web Data Compression and Search and COMP6714 (http://www.handbook.unsw.edu.au/postgraduate/courses/2017/COMP6714.html) Information Retrieval and Web Search.

Learning Outcomes

By the end of the course, you should be able to:

- develop accurate, non-redundant data models;
- realise data models as relational database schemas;
- formulate queries via the full range of SQL constructs;
- use stored procedures and triggers to extend DBMS capabilities;
- understand principles and techniques for administering RDBMSs;
- understand performance issues in relational database applications;
- understand the overall architecture of relational DBMSs;
- understand the concepts behind transactions and concurrency control;
- appreciate query and transaction processing techniques within RDBMSs;
- appreciate the past, present and future of database technology.

Glossary:

- **DBMS** : DataBase Management System ... software system to support database manipulation.
- **RDBMS** : Relational DBMS ... the most popular style of DBMS (refers to underlying data model) .
- **SQL** : Structured Query Language ... the ANSI standard language for manipulating RDBMSs.

This course contributes to the development of the following graduate attributes:

Graduate Capability	**Acquired in**
Scholars capable of independent and collaborative enquiry, rigorous in their analysis, critique and reflection, and able to innovate by applying their knowledge and skills to the solution of novel as well as routine problems	Lab work, Assignments
Entrepreneurial leaders capable of initiating and embracing innovation and change, as well as engaging and enabling others to contribute to change	Lab work, Assignments
Professionals capable of ethical, self- directed practice and independent lifelong learning	Lab work, Assignments
Global citizens who are culturally adept and capable of respecting diversity and acting in a socially just and responsible way	Lab work, Assignments

Assumed Knowledge

There is no official pre-requisite. However, COMP9021 is the course co-requisite. Therefore, before/during this course, students should/will have:

- experience with procedural programming; and
- an understanding of a range of in-memory searching data structures (e.g., binary search trees, and hash-tables).

Teaching Strategies

- Lectures: introduce concepts and show examples
- Lab Work: reinforce concepts, provide additional examples, and allow students to solve problems
- Assignments: further reinforce concepts and allow students to solve significant problems

Teaching Rationale

Since this course is about the understanding and effective use of a specific technology (relational database management systems), practical use of the technology is critical to the learning outcomes of the course. Thus, a lot of time will be spent during lectures to work through exercises, to illustrate the practice of using various database techniques and technologies. However, the primary learning focus in this course is assignment work, which has been designed to be challenging and relevant (i.e., dealing with real problems and using real database schemas).

Student Conduct

The **Student Code of Conduct** (Information (https://student.unsw.edu.au/conduct) , Policy (https://www.gs.unsw.edu.au/policy/documents/studentcodepolicy.pdf)) sets out what the University expects from students as members of the UNSW community. As well as the learning, teaching and research environment, the University aims to provide an environment that enables students to achieve their full potential and to provide an experience consistent with the University's values and guiding principles. A condition of enrolment is that students *inform themselves* of the University's rules and policies affecting them, and conduct themselves accordingly.

In particular, students have the responsibility to observe standards of equity and respect in dealing with every member of the University community. This applies to all activities on UNSW premises and all external activities related to study and research. This includes behaviour in person as well as behaviour on social media, for example Facebook groups set up for the purpose of discussing UNSW courses or course work. Behaviour that is considered in breach of the Student Code Policy as discriminatory, sexually inappropriate, bullying, harassing, invading another's privacy or causing any person to fear for their personal safety is serious misconduct and can lead to severe penalties, including suspension or exclusion from UNSW.

If you have any concerns, you may raise them with your lecturer, or approach the School Ethics Officer (mailto:ethics-officer@cse.unsw.edu.au) , Grievance Officer (mailto:grievance-officer@cse.unsw.edu.au) , or one of the student representatives (mailto:stureps.ug@cse.unsw.edu.au) .

Plagiarism is defined as (https://student.unsw.edu.au/plagiarism) using the words or ideas of others and presenting them as your own. UNSW and CSE treat plagiarism as academic misconduct, which means that it carries penalties as severe as being excluded from further study at UNSW. There are several on-line sources to help you understand what plagiarism is and how it is dealt with at UNSW:

- Plagiarism and Academic Integrity (https://student.unsw.edu.au/plagiarism)

- UNSW Plagiarism Procedure (https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf)

Make sure that you read and understand these. Ignorance is not accepted as an excuse for plagiarism. In particular, you are also responsible that your assignment files are not accessible by anyone but you by setting the correct permissions in your CSE directory and code repository, if using. Note also that plagiarism includes paying or asking another person to do a piece of work for you and then submitting it as your own work.

UNSW has an ongoing commitment to fostering a culture of learning informed by academic integrity. All UNSW staff and students have a responsibility to adhere to this principle of academic integrity. Plagiarism undermines academic integrity and is not tolerated at UNSW. Plagiarism at UNSW is defined as using the words or ideas of others and passing them off as your own.

If you haven't done so yet, please take the time to read the full text of

- UNSW's policy regarding academic honesty and plagiarism (https://student.unsw.edu.au/plagiarism)

The pages below describe the policies and procedures in more detail:

- Student Code Policy (https://www.gs.unsw.edu.au/policy/documents/studentcodepolicy.pdf)
- Student Misconduct Procedure (https://www.gs.unsw.edu.au/policy/documents/studentmisconductprocedures.pdf)
- Plagiarism Policy Statement (https://www.gs.unsw.edu.au/policy/documents/plagiarismpolicy.pdf)
- Plagiarism Procedure (https://www.gs.unsw.edu.au/policy/documents/plagiarismprocedure.pdf)

You should also read the following page which describes your rights and responsibilities in the CSE context:

- Essential Advice for CSE Students (https://www.engineering.unsw.edu.au/computer-science-engineering/about-us/organisational-structure/student-services/policies/essential-advice-for-cse-students)

Assessment

Task	Released	Due	Weighting
Assignment 1	Week 2	Week 5	10%
Assignment 2	Week 5	Week 8	15%
Assignment 3	Week 8	Week 11	15%
Final Exam	Exam Period	Exam Period	60%

Late Assignment

If an assignment is handed in after the due date, a 10% penalty will be applied to the total mark for each day (or part thereof) late after the due date (including weekends and public holidays).

The Final Exam in this course will take place during exam period. It will be a 2 hour exam.

Final Mark

Your final mark for this course will be computed using the above assessments as follows:

FinalMark	=	Assignment1Mark + Assignment2Mark + Assignment3Mark + FinalExamMark	out of 100
ExamOK	=	FinalExamMark ≥ 24/60	true/false
FinalGrade	=	UF, if !ExamOK && FinalMark > 50/100 FL, if FinalMark < 50/100 PS, if 50/100 ≤ FinalMark < 65/100 CR, if 65/100 ≤ FinalMark < 75/100 DN, if 75/100 ≤ FinalMark < 85/100 HD, if FinalMark ≥ 85/100	

Students are eligible for a Supplementary Exam if and only if:

- they cannot attend the final exam due to illness or misadventure
- their final mark is in the range 47 ≤ FinalMark < 50 (in this case, FinalMark is limited to 50)

A Supplementary Exam will not be awarded for any other reason.

Course Schedule

The following is an **approximate** guide to the sequence of topics in this course.

Week	Lectures	Assignments
1	Introduction, Data Modelling, ER Notation	
2	Relational Model, ER-Relational Mapping, SQL Schemas	
3	DBMSs, Databases, Data Modification	
4	SQL Queries	
5	More SQL Queries, Stored Procedures, PLpgSQL	Assignment 1 Due
6	Extending SQL: Queries, Functions, Aggregates, Triggers	
7	More Triggers, Programming with Databases	
8	Catalogs, Privileges	Assignment 2 Due
9	Relational Design Theory, Normal Forms	
-	Non-teaching week (mid-semester break)	-
10	Relational Algebra, Query Processing	
11	Transaction Processing, Concurrency Control	Assignment 3 Due
12	Course Review	

Resources for Students

Textbook:

- Fundamentals of Database Systems (http://wps.aw.com/aw_elmasri_fundatasys_7) , Elmasri and Navathe, 7th edition, 2016, Addison-Wesley

Reference books:

- Database System Concepts (http://db-book.com/) , Silberschatz, Korth, Sudarshan, 6th edition, 2010, McGraw-Hill
- Database Management Systems (http://www.cs.wisc.edu/~dbbook/) , Ramakrishnan and Gehrke, 3rd edition, 2003, McGraw-Hill
- Database Systems: The Complete Book (http://www-db.stanford.edu/~ullman/dscb.html) , Garcia-Molina, Ullman, Widom, 2nd edition, 2008, Prentice-Hall
- Database Systems: An Application-Oriented Approach (http://wps.aw.com/aw_kifer_datacomp_2) Kifer, Berstein, Lewis, 2nd edition (Complete Version), 2006, Addison-Wesley

If you have access to an earlier edition of any of these (one or two editions less than the one given above), it'll be fine for this course.

The textbook and reference books give the greatest detail on topics covered in the course. The Lecture Slides are less detailed, but have the advantage that they cover precisely the syllabus (but no more). They also contain examples that will be discussed in detail. All Lecture Slides and solutions to the examples from lectures will be posted on the course web site.

The software system to be used in this course is PostgreSQL. The documentation and manuals (http://www.postgresql.org/docs/) provided with PostgreSQL are actually very good, so you don't need to buy textbooks for these. However, if you feel more comfortable with a book, there are references to a range of books on the web site for PostgreSQL (http://www.postgresql.org/docs/books/) .

A general problem with technology textbooks is that they go out-of-date very quickly. Another problem is that many of them provide a brief introduction with some examples, and then give a summary of the manual.

PostgreSQL will be used as the primary DBMS in this course because it is a typical example of a full-featured client-server DBMS, and has the added bonuses that (a) it has a simple extensibility model and (b) has the source code available if you want to learn more about how DBMSs work. Commercial alternatives could have been Oracle, DB2, SQLserver, although none of these are available in source code form. (MS Access is not a full-featured relational database.) The only plausible open-source alternative is MySQL, but it was not a full-featured DBMS until version 5, and has a source code base that is largely cobbled together from a number of existing open-source systems. The PostgreSQL code base, on the other hand, is the result of coherent development by a relatively small team.

Course Evaluation and Development

This course is evaluated each session using the myExperience system at the end of the semester.

The evaluation from the previous offering of the course (17s1) showed that students were generally satisfied with the course. Based on their comments, new lab demonstrators will be used this semester. Therefore, your feedback is important and will be considered to improve future offerings of this course.

Students are also encouraged to provide informal feedback during the semester, and let the lecturer in charge know of any problems, as soon as they arise. Suggestions will be listened to very openly, positively, constructively and thankfully, and every reasonable effort will be made to address them as soon as possible.

Resource created 4 years ago (Friday 07 July 2017, 12:19:56 PM), last modified 4 years ago (Sunday 23 July 2017, 08:07:51 PM).

Comments (/COMP9311/17s2/forums/search?forum_choice=resource/9361) (/COMP9311/17s2/forums/resource/9361)

There are no comments yet.