

## COMP9321 – Data Services Programming

Created: 30 Apr 2017

Proposal Last Updated: 02 May 2017

## Offering Details:

## Key Details and Contacts

## Key Course Details

Course Name (Official)	Data Services Programming		
Standard Name (SIMS)	Data Services Programming		
Course Code	COMP9321		
Units of Credit (UOC)	6		
Career	Hybrid		
Course Offerings	<b>Offering Number</b>	<b>Career</b>	<b>Course Code</b>
	1	PG	COMP9321
	2	UG	COMP9321
Level	3		
First semester and year the revised changes will take effect	2018 Semester 1		

## Contact Details

Proposal Proponent	<b>Name</b>	<b>Email</b>	<b>Role</b>
	Boualem Benatallah	boualem@cse.unsw.edu.au	Professor, School of Computer Science and Engineering
Proposal Author(s)	<b>Name</b>	<b>Email</b>	<b>Role</b>
	Hye-Young Paik	hpaik@unsw.edu.au	Senior Lecturer, School of Computer Science and Engineering
Proposal Contact	<b>Name</b>	<b>Email</b>	<b>Role</b>
	Hye-Young Paik	hpaik@unsw.edu.au	Senior Lecturer, School of Computer Science and Engineering
Optional Additional Endorsers	Not specified		
Academic Unit responsible for course	School of Computer Science and Engineering		
Parent Academic Unit	Faculty of Engineering		

## Proposal Concept

## Summary of Proposal

Summary of Proposal	In this revision, we aim to shift the focus of the course content from generic Web application design and implementation methodology to Web data access and analytics. The core concepts of the Web application such as design patterns, basic scalability issues for Web apps are still taught, but they will be presented in the context of consuming, analysing and visualising Web data.
---------------------	--

## Justification for proposal

Justification for Proposal	<p>Web application design and development has a fast-changing landscape. With the continuous improvement in connectivity, storage and data processing capabilities, characteristics and functionality of a modern Web application is often defined by its ability to access a range of public/private data from various sources. Therefore, well-engineered service-oriented functionalities are critical for ingesting, organizing and querying the growing volume of data in data-driven Web applications.</p> <p>There is an identified gap in current curriculums (in CSE, and Data Science and Decisions program in SMS) in terms of teaching core aspects of managing data to support high-level data analysis techniques. Most courses focus on the breadth and depth of core techniques (e.g. data mining algorithms, statistical methodologies), there are no topics that discuss how to design and build services and applications using those techniques. The revision aims to fill the gap by offering Web-based methodologies for data intensive processing and programming.</p>
----------------------------	---

## Attachments

Attach documentation to this proposal	None attached
---------------------------------------	---------------

## Learning and Teaching

### Learning & Teaching development and support

Are there Learning & Teaching space requirements for the course beyond those that can be accommodated by CATS spaces?

No

Have you discussed with the Learning Centre and Learning and Teaching what language and/or academic skills development resources and/or which teaching and learning strategies might be suited to this course?

No

Are many students in this course at a key transition point where their academic skills are likely to need development, e.g. from one kind of educational institution or type of program to another or into education after a significant break?

No

## Consultation

### Internal consultation

#### Internal Consultation

**Consultants**

None specified

**Details**

Discussed within the service oriented computing group and other relevant academic in CSE. We also have looked into the new program Bachelor of Data Science and Decisions (school of Math).

**Attachments**

None specified

### External consultation

#### External Consultation

**Consultants**

None specified

**Details**

Prof. Fabio Casati (University of Trento, Italy) provided input to the revised curriculum based on his expertise and experience in teaching similar courses.

**Attachments**

None specified

#### Interested Parties

Not specified

## Related Proposals

### Related Proposals

Code	Proposal Name	Type	Date	Status
COMP9322	Software Service Design and Engineering	Course Revision (HY)	Apr 2017	Submitted

## Endorsements and Comments

### Endorsement history

No endorsements have been recorded for this proposal (yet).

### Comments

No comments posted

Administration:

**Key Course Details**

**Key Admin Details**

Course Name (Official)	Data Services Programming
Student System ID	00005853
Can course be taken as General Education elective?	No
Field of Education	020399 – Information Systems not elsewhere classified

**Course Review**

Next course review date	July 01, 2022
Provide details of any particular factors that need to be considered at that review.	Making sure that the technologies used in the course and lab/assignment activities are up-to-date.

**Delivery and Attendance**

Campus administering the Course	Sydney
---------------------------------	--------

Teaching Shares by School/Faculty	<b>School</b>	<b>Teaching Share (%)</b>
	School of Computer Science and Engineering	100
	<b>Total Share</b>	<b>100</b>

Semesters the course is offered		<b>Summer Semester</b>	<b>Semester 1</b>	<b>Semester 2</b>
	2017	No	No	No
	2018	No	Yes	Yes
	2019	No	Yes	Yes
	2020	No	Yes	Yes

Teaching mode and contact hours	Standard Offering Mode
---------------------------------	------------------------

Standard offering contact hours per week	<b>Learning Activity</b>	<b>Hours/Week</b>
	Lecture	3
	Tutorial/Laboratory	0
	Tutorial	0
	Laboratory	0
	Web-based Online Learning Activity	2
	Clinical/Fieldwork	0
	Distance Learning	0
	Seminar	0
	Studio	0
	Meeting/Consultation	0
<b>Total Hours per week</b>	<b>5</b>	

Primary delivery mode	Classroom
-----------------------	-----------

Secondary delivery modes	Online
--------------------------	--------

Additional information about the delivery modes for this course	This course will mainly be delivered through a mix of face-to-face lectures, online videos, self-guided online lab activities and online forum (Q/A).
---	---

**Staff**

**Staff associated with course**

Course Convenor	Name	Email	Role
	Boualem Benatallah	boualem@cse.unsw.edu.au	Professor, School of Computer Science and Engineering
	Fethi Rabhi	f.rabhi@unsw.edu.au	Professor, School of Computer Science and Engineering
	Hye-Young Paik	hpaik@unsw.edu.au	Senior Lecturer, School of Computer Science and Engineering
Administrative Contact	Name	Email	Role
	Hye-Young Paik	hpaik@unsw.edu.au	Senior Lecturer, School of Computer Science and Engineering

## Supplementary Information:

### Resources

#### Student Resources

##### Prescribed Resources

<b>1.</b>	<b>Lecture notes</b>	<b>Other</b>
<b>Resource Type</b>	Written lecture notes	
<b>Additional Details</b>	Lecture notes are provided for each topic (weekly)	
<b>2.</b>	<b>Lab notes</b>	<b>Other</b>
<b>Resource Type</b>	Written lab notes	
<b>Additional Details</b>	Lab notes for the practical weekly activities	
<b>3.</b>	<b>COMP9321 Course Homepage</b>	<b>Website</b>
<b>URL</b>	<a href="http://www.cse.unsw.edu.au/~cs9321">http://www.cse.unsw.edu.au/~cs9321</a>	
<b>Publisher</b>	School of Computer Science and Engineering	
<b>Additional Details</b>	Course homepage (updated every session)	

##### Recommended Resources

None specified

### Experience and Assumed Knowledge

#### Industrial Experience Component

##### Industrial Experience Component

Not specified

#### Assumed Knowledge

##### Assumed Knowledge

Not specified

## Academic Structure:

### Academic Structure

#### Prerequisites

Prerequisite courses	Not specified
Prerequisite programs	Not specified
Prerequisite streams	Not specified
Prerequisite conditions	For Undergrad: COMP1531, COMP2041. For Postgrad: COMP9021,COMP9311

#### Exclusions

Excluded Courses	Not specified
Excluded Programs	Not specified
Excluded Streams	Not specified

#### Equivalent

Equivalent courses	Not specified
--------------------	---------------

### Assessment

#### Assessment

Grading Basis	Standard UNSW grades (e.g. HD, DN, CR, PS, FL)
---------------	--

#### Assessment items and their relationship to Course Learning Outcomes

Assessment Title	Assessment Type	Weight (%)
1 Programming Assignments	Assignment	40%
Assessment Description:	A programming assignment tackling a non-trivial problem. This will assess the student's ability to apply the learned theory of Data services in a practical scenario. The assignment will be marked against a clearly specified marking criteria.	
2 Quizzes	Test	10%
Assessment Description:	Online quizzes on lecture topics. The quizzes are automatically marked. The answers to the quizzes will be discussed either through the course forum or during lecture times.	
3 Final Exam	Examination	50%
Assessment Description:	A formal written exam at the end of the course.	
<b>Total Weight</b>		<b>100%</b>

#### Programming Assignments

- Understand the fundamentals of data visualisation and how to communicate effectively with data
- Design and develop a non-trivial Data Services to access, ingest, curate and analyse the data

#### Quizzes

- Describe the main requirements to design and implement APIs (specifically REST APIs)
- Describe the main requirements of data-driven applications
- Identify and apply Design Patterns in data-driven applications
- Understand the fundamentals of data visualisation and how to communicate effectively with data
- Design and develop a non-trivial Data Services to access, ingest, curate and analyse the data

#### Final Exam

- Understand the fundamentals of data visualisation and how to communicate effectively with data
- Design and develop a non-trivial Data Services to access, ingest, curate and analyse the data
- Identify and apply Design Patterns in data-driven applications
- Describe the main requirements of data-driven applications
- Describe the main requirements to design and implement APIs (specifically REST APIs)

### Curriculum Mapping

#### Course Learning Outcomes

<b>Specify the learning outcomes that students should achieve upon successful completion of this course</b>	<ol style="list-style-type: none"> <li>1 Describe the main requirements to design and implement APIs (specifically REST APIs)</li> <li>2 Describe the main requirements of data-driven applications</li> <li>3 Identify and apply Design Patterns in data-driven applications</li> <li>4 Design and develop a non-trivial Data Services to access, ingest, curate and analyse the data</li> <li>5 Understand the fundamentals of data visualisation and how to communicate effectively with data</li> </ol>
<b>Teaching strategies and Rationale</b>	
<b>Teaching Strategies and Rationale</b>	<p>The course is designed to encourage the students to learn by doing. We provide timely feedback for learning via small, step-by-step weekly practical activities and tests that gradually build up knowledge and practical skills.</p> <p>Through face-to-face lectures and online videos, we introduce concepts and theory with practical examples. In lab work, we introduce the technologies and techniques for the assignments. The assignments allow students to solve significant problems.</p>
<b>Course Aims</b>	
<b>Course Aims</b>	<p>Software engineering has advanced rapidly in recent years. The knowledge-, service-, and cloud-based economy in parallel with the continuous improvement in connectivity, storage and data processing capabilities allow access to a data deluge from sensors, social-media, news, user-generated, government and private data sources. Accordingly, in a modern data-oriented landscape, data-driven applications may need to deal with a collection of datasets - from relational to NoSQL - that holds a vast amount of data gathered from various private/open data islands. Therefore, well-engineered service-oriented functionalities are critical for ingesting, organizing and querying the growing volume of data in data-driven application.</p> <p>This course aims to introduce the student to core concepts and practical skills for engineering the data in service-oriented data-driven applications. Specifically, the course aims to answer these questions:</p> <ul style="list-style-type: none"> <li>• How to develop services to access and ingest data in internal/external sources of the data?</li> <li>• How to develop services to use Databases (from Relational to NoSQL) as a Service for persisting user information?</li> <li>• How to develop services to Curate (e.g. Extract, Transform, Correct, Aggregate, and Merge/Split) the data?</li> <li>• How to develop services to apply analytics (e.g. by leveraging Machine Learning and Natural Language Processing techniques) to the curated data?</li> <li>• How to develop services to visualize the data to communicate effectively with data.</li> </ul>

---

## Publications and Marketing:

### Publications

#### Course Description

**Description of course that can be used in online publications (e.g. Handbook website, Faculty websites or other online catalogue systems)**

This course aims to introduce the student to core concepts and practical skills for engineering the data in Web service-oriented data-driven applications. Specifically, the course aims to expose students to basic infrastructure for building data services on the Web, including techniques to access and ingest data in internal/external sources, develop software services to curate (e.g. extract, transform, correct, aggregate the data, develop services to apply various analytics and develop services to visualize the data to communicate effectively with data. The course uses the Python Programming Language as the practical basis for its modules. However, the concepts taught are universal and can be applied to any other web development framework.

#### Key Search Terms

**List key search terms that might be used to search for this course (e.g. via the Handbook or Google searches).**

Data services  
Software services  
Data analytics  
Web applications  
Software engineering