

School of Information Management

INFO411 INFORMATION SYSTEMS ANALYSIS AND DEVELOPMENT

Trimester 2, 2015

COURSE OUTLINE

Names and Contact Details

Dr. Yi-Te Chiu

Room: RH412, Rutherford House

E-mail: yi-te.chiu@vuw.ac.nz

Phone: 04-463-5689

Office hours: by appointment

Trimester Dates

Teaching Period: Monday 13th July – Friday 30 October

Withdrawal from Course

1. Your fees will be refunded if you withdraw from this course on or before Friday 24th July 2015.
2. The standard last date for withdrawal from this course is Friday 25th September. After this date, students forced to withdraw by circumstances beyond their control must apply for permission on an '*Application for Associate Dean's Permission to Withdraw Late*' form including supporting documentation. The application form is available from either of the Faculty's Student Customer Service Desks or [online](#).

Class Times and Room Numbers

Time: Wednesday 9:30 am - 12:20 pm

Room: Railway West Wing (RWW126)

Course Delivery

Each class will be run as a seminar. You must read all required articles prior to class and be prepared to discuss them. Participating includes offering new and unique insights, clarifying issues and complexities, reframing and extending ideas in meaningful ways, and offering a perspective that helps the group integrate and synthesize readings, ideas, and topics. Moreover, I aim to create an environment where all participants in the seminar learn from each other. Sharing your own ideas without regard for colleagues' insights is not sufficient. Participation means that you should listen carefully to your colleagues' ideas, and then thoughtfully build upon them.

- Discussion facilitator

Each of you will be required to be the discussion topic facilitator for two sessions (first on pair and later on your own) during the course. After carefully reviewing the course outline, you will be asked to submit your top three choices by the end of our first session. I will do my best to accommodate your requests.

The discussion facilitator is responsible for developing a creative class structure that engages class members while facilitating learning. You are responsible for getting the group to engage in a critical yet constructive discussion of the key issues, challenges and dilemmas raised in the readings. Begin with a brief review of the key points and issues raised in the required articles (no more than 10 minutes). The goal is to set the foundation for the discussion by highlighting key points, not to provide an exhaustive review of everything said in the readings. Assume that everyone has read all articles. You should be creative in designing a session that will stimulate dialogue, interactions and the creation of knowledge. You should aim for about 80 minutes of facilitated discussion. I would strongly recommend that you prepare discussion questions that will engage the class, engender debate, encourage synthesis, and dialogue. Discussion questions that do not have “correct” answers are typically the most beneficial in this regard. You may also want to develop other activities that will encourage the class to integrate across ideas. The followings are few ideas:

- You can have the group synthesize ideas into an integrative model
- You can set up a debate to identify the key dilemmas in the literature
- You can identify interesting cases relevant to topics and encourage the application of key concepts from literature in case discussion.
- You can encourage the group make cross-topic connections between the current readings and those in prior sessions.

Overall, your facilitation should demonstrate that you have become a subject matter expert on the topic at-hand, and that you have thought very carefully about the types of questions and activities that will stimulate learning based upon the readings.

▪ Thought items

For each session, except for the discussion facilitator, each person will need to submit a thoughtful analysis of the required readings for that week. A thought item is a short demonstration of your ability to think creatively and integratively about concepts covered in the course. A thought item is *NOT* article summaries. These thought items can take numerous different forms. You can write a thought item by answering following questions or you can write a thought item showing your critical and creative thinking.

- What are the key issues, dilemmas, and/or controversies raised in this set of readings? How might you see resolving them?
- What exciting advances in practices, research, and/or theory appealed to you from the readings? Why? How might you apply to practical problems?
- What are the key research questions that you think need to be answered based upon this set of readings? How would you go about pursuing them?

Each thought item should be 1-2 page double-spaced typed pages (with 1” margins, Times New Roman 12-point font). They should be e-mailed to the course coordinator by 6 pm each Tuesday. You should also submit a hard-copy to me in class. *Late thought items will not be accepted.*

Group Work

Students are required to participate in a group presentation as part of ISD methodology report. This will involve an average of 10 hours work outside the class room.

Expected Workload

This is a 15-point course. One point equates to approximately 10 hours of work, for a total of 150 hours for the course. With 12 three-hour classes, a total of 36 in-class hours are required. The remaining 114 hours will be spread over the 12 teaching weeks and the mid-trimester break. The

following breakdown estimates the required time for each task, giving you a rough idea of how much time you may need to spend.

- Class preparation: 64 hours
- ISD methodology report and review paper: 40 hours
- Test: 10 hours

Prescription

Advanced topics in systems analysis and development. Topics may include: system development methodologies, the human aspects of system development, dynamics of system development teams, requirements elicitation & analysis, system design, the management of system development projects, the evaluation of system development, and system maintenance.

Course Learning Objectives

1. Recognize the complexity of system development;
2. Evaluate leading-edge trends and their implications for systems development;
3. Compare and contrast diverse system development methodologies, tools, techniques, and practices;
4. Describe and apply theoretical perspectives and concepts to diverse challenges that may arise during system development;
5. Identify the crucial factors for successful system development;
6. Apply essential problem-solving skills for system development.

Course Content

Information Systems (IS) plays a crucial function in organizations, as revealed by significant organizational expenditures on IT (4.3% of revenue, Gartner Research, 2010). Many organizations rely on IS to improve their business productivity and to gain a competitive advantage over their competitors. Yet, despite its value, many organizations suffer from high failure rates in their IS projects, which cost multi-billion dollars per year in the U.S. (Standish Group, 2009). The difficulty in successfully implementing IS largely stems from the complexity of information systems development (ISD), which involves not only the management of multiple technical components, but also that of people and change.

These challenges are addressed in this course through an in-depth examination of the general literature and selected case studies. We will integrate theory and practice to understand and improve system development. This course covers historically important, foundational papers as well as cutting-edge articles that reflect the latest IS research. We will compare and contrast a variety of perspectives from prior literature. By the end of the course, you will likely recognize that there is no single method to resolve ISD issues, and will be able to apply the acquired knowledge to successfully manage ISD.

Week	Class	Topic
1	15 July	<ul style="list-style-type: none"> • Course introduction: Topics and assessments • Challenges in ISD • Human aspects of ISD
2	22 July	System development methodologies (1): Foundation and evolving schools of thought
3	29 July	System development methodologies (2): Adoption and implementation
4	5 August	System analysis & design (1): Analysts and requirements management
5	12 August	System analysis & design (2): New challenges in analysis and design
6	19 August	System development teams (1): Team composition and processes

		Mid-trimester break (8/24–9/4) *A one-page proposal for the topic of your review paper is due by 5 pm on 4 September.
7	9 September	(1) Student presentations (System development methodology) (2) Proposal discussion *Written report due by 5 pm on 15 September.
8	16 September	System development teams (2): Team knowledge and team management
9	23 September	System development teams (3): The Impacts of ICTs
10	30 September	System development projects (1): Control and commitment
11	7 October	System development projects (2): Performance management, success, and failures
12	14 October	Global system development: Offshoring and open source development
13	21 October	Final Test

Readings

There is no textbook for this class. A list of readings for each week is attached in the end of the course outline. All materials will be available for download from Blackboard, or from Internet links.

Assessment

Requirement	Due Date	Weight
ISD methodology report (LO 2, 3, and 5) You will work with one of your colleagues for this assignment. Each team will need to: (1) present a selected ISD methodology (5%) and (2) write an in-depth analysis of a selected methodology (10%)	(1) 9 September (in-class) (2) 15 September before 5pm (by e-mail)	15%
Review Paper (LO 1–6) The review paper allows you to further explore a topic of interest from the course content. (1) A one-page proposal for your topic. (Not assessed) (2) Review paper (30%)	(1) 4 September before 5 pm (by e-mail) (2) 30 October before 5 pm (by e-mail)	30%
Test (LO 1–6) The final test will cover all class material. This will be one-day take-home test. You will receive test questions 9 am on the test date by email. You need to return submission by <u>4 pm on the same date</u> by email. The test may consist of short essays.	October 21	20%
Seminar leadership (LO 1, 2, 4, and 6)	Varied by individual schedule	20%
Class contribution and thought items (LO 1-6) (1) Participation consists of listening carefully to your colleagues' ideas, and then thoughtfully building upon them. It also includes offering new and unique insights, clarifying issues and complexities, reframing and extending ideas in meaningful ways, and offering a perspective that helps the group integrate and synthesize readings, ideas, and topics. (2) Thought items	(1) continuous (2) 6 pm each Tuesday	15%

Note: Detailed assignment guidelines will be provided on/through Blackboard and in class.

If you cannot complete an assignment or sit a test or examination, refer to www.victoria.ac.nz/home/study/exams-and-assessments/aegrotat

The Assessment Handbook will apply to all VUW courses: see <http://www.victoria.ac.nz/documents/policy/staff-policy/assessment-handbook.pdf>.

Indicative System Development Methodology Topics

- Soft Systems Methodology
- Capability Maturity Model
- Rational Unified Process
- Rapid Application Development
- Structured Analysis, Design, and Implementation of Information Systems (STRADIS)
- Scrum
- eXtreme Programming (XP)
- Lean software development
- Or methodology of your choice negotiated with the course co-ordinator

Penalties

In fairness to other students, assignment work submitted after the deadline will incur a 10% penalty for *each* 24-hour period past the deadline. In the event of bereavement or prolonged illness affecting your ability to meet the deadline, discuss your situation with the Course Co-ordinator. You must verify your claim, e.g., produce a medical certificate. In doing so, you consent to your supporting documentation being checked by the Course Co-ordinator. Extensions will only be granted under these conditions.

Use of Turnitin

Student work provided for assessment in this course may be checked for academic integrity by the electronic search engine <http://www.turnitin.com>. Turnitin is an on-line plagiarism prevention tool which compares submitted work with a very large database of existing material. At the discretion of the Head of School, handwritten work may be copy-typed by the School and submitted to Turnitin. A copy of submitted materials will be retained on behalf of the University for detection of future plagiarism, but access to the full text of submissions will not be made available to any other party.

Mandatory Course Requirements

None.

Student feedback

Student feedback on University courses may be found at www.cad.vuw.ac.nz/feedback/feedback_display.php

Link to general information

For general information about course-related matters, go to <http://www.victoria.ac.nz/vbs/studenthelp/general-course-information>

Note to Students

1. Your assessed work may also be used for quality assurance purposes, such as to assess the level of achievement of learning objectives as required for accreditation and academic audit. The findings may be used to inform changes aimed at improving the quality of VBS programmes. All material used for such processes will be treated as confidential, and the outcome will not affect your grade for the course.
2. This course syllabus provides a general plan for the course; deviations may be necessary.

INFO 411 Reading List¹

Week 1 Introduction

(a) Challenges in ISD

1. Bostrom, R. P., & Heinen, J. S. (1977). MIS problems and failures: A socio-technical perspective—part I: the causes. *MIS Quarterly*, 1(3), 17–32.
2. Doherty, N. F., & King, M. (1998). The consideration of organizational issues during the systems development process: an empirical analysis. *Behaviour & Information Technology*, 17(1), 41–51.
3. Xia, W., & Lee, G. (2004). Grasping the complexity of IS development projects. *Communications of the ACM*, 47, 68–74.

(b) Human Aspects of System Development

1. Sharp, H., Baddoo, N., Beecham, S., Hall, T., & Robinson, H. (2009). Models of motivation in software engineering. *Information and Software Technology*, 51(1), 219–233.

Week 2: System Development Methodologies (1): Foundation and Evolving Schools of Thought

1. Basics of System Development Methodologies: Chapter 27 (p.567-590) in Avison, D. E., & Fitzgerald, G. (2007). *Information systems development: methodologies, techniques and tools* (4th ed.). New York, NY: McGraw-Hill.
2. Alter, S. (2001). Which Life Cycle–Work System, Information System, or Software?. *Communications of the Association for Information Systems*, 7(1), 1–53. (focus on pp.1-18)
3. Glass, R. L. (2001). Agile versus traditional: Make love, not war!. *Cutter IT Journal*, 14(12), 12–18.
4. Glazer, H., Dalton, J., Anderson, D., Konrad, M. D., & Shrum, S. (2008). CMMI or Agile: Why not embrace both! (Technical Note No. CMU/SEI-2008-TN-003). Software Engineering Institute.
5. [Case discussion] Austin, R. D., Nolan, R. L., & O'Donnell, S. (2009). Project Management: What's the Best Approach for IT? In *The adventures of an IT leader* (pp. 89–106). Boston, MA: Harvard Business Press.

[Supplementary]

1. For those who are not familiar with system development cycle, read Chapter 3 in Avison, D. E., & Fitzgerald, G. (2007). *Information systems development: methodologies, techniques and tools* (4th ed.). New York, NY: McGraw-Hill.
2. Agile manifesto <http://agilemanifesto.org/>

Week 3: System Development Methodologies (2): Adoption and Implementation

1. Barlow, J. B., Keith, M. J., Wilson, D. W., Schuetzler, R. M., Lowry, P. B., Vance, A., & Giboney, J. S. (2011). Overview and Guidance on Agile Development in Large Organizations. *Communications of the Association for Information Systems*, 29, 25–44.

¹ Please note that *readings are subject to change*, including addition/deletion, at the instructor's discretion. Check the BlackBoard Website for latest weekly readings.

2. Edberg, D., Ivanova, P., & Kuechler, W. (2012). Methodology mashups: An exploration of processes used to maintain software. *Journal of Management Information Systems*, 28(4), 271–304.
3. Iivari, J., & Iivari, N. (2011). The relationship between organizational culture and the deployment of agile methods. *Information and Software Technology*, 53(5), 509–520.
4. Fitzgerald, B., Russo, N. L., & O’Kane, T. (2003). Software development method tailoring at Motorola. *Communications of the ACM*, 46(4), 64–70.

[Supplementary]

1. Austin, R. D., & Devin, L. (2009). Research commentary-weighing the benefits and costs of flexibility in making software: Toward a contingency theory of the determinants of development process design. *Information Systems Research*, 20(3), 462–477.

Week 4 System Analysis & Design (1): Analysts and Requirements Management

1. Business Analysts and System Analysts: Chapter 2-4, 6, and 11 in Blais, S. (2011). *Business analysis: best practices for success*. Hoboken, NJ: John Wiley & Sons.
2. Browne, G. J., & Ramesh, V. (2002). Improving information requirements determination: a cognitive perspective. *Information & Management*, 39(8), 625–645.
3. Markus, M. L., & Mao, J.-Y. (2004). Participation in development and implementation-updating an old, tired concept for today’s IS contexts. *Journal of the Association for Information Systems*, 5(11-12), 515–544.

Week 5 System Analysis & Design (2): New Challenges in Analysis and Design

1. Sommerville, I. (2005). Integrated requirements engineering: A tutorial. *Software, IEEE*, 22(1), 16–23.
2. Cao, L., & Ramesh, B. (2008). Agile requirements engineering practices: An empirical study. *Software, IEEE*, 25(1), 60–67.
3. Elfatry, A. (2007). Dealing with change: components versus services. *Communications of the ACM*, 50(8), 35–39.
4. Lichtenstein, S., Nguyen, L., & Hunter, A. (2007). Issues in IT service-oriented requirements engineering. *Australasian Journal of Information Systems*, 13(1), 176–191.

Week 6 System Development Teams (1): Team Composition and Processes

1. Gorla, N., & Lam, Y. W. (2004). Who should work with whom? Building effective software project teams. *Communications of the ACM*, 47(6), 79–82.
2. Liang, T. P., Jiang, J., Klein, G. S., & Liu, J. Y. C. (2010). Software quality as influenced by informational diversity, task conflict, and learning in project teams. *Engineering Management, IEEE Transactions on*, 57(3), 477–487
3. Tiwana, A., & Mclean, E. R. (2005). Expertise integration and creativity in information systems development. *Journal of Management Information Systems*, 22(1), 13-43.
4. Pendharkar, P. C., & Rodger, J. A. (2009). The relationship between software development team size and software development cost. *Communications of the ACM*, 52(1), 141–144.
5. Teh, A., Baniassad, E., Van Rooy, D., & Boughton, C. (2012). Social psychology and software teams: Establishing task-effective group norms. *Software, IEEE*, 29(4), 53–58.
6. Akgün, A. E., Keskin, H., Cebecioglu, A. Y., & Dogan, D. (2015). Antecedents and consequences of collective empathy in software development project teams. *Information & Management*, 52(2), 247–259.

[Supplementary]

1. Kozlowski, S. W., & Ilgen, D. R. (2006). Enhancing the effectiveness of work groups and teams. *Psychological Science in the Public Interest*, 7(3), 77–124.

Week 7: ISD methodology presentation (no reading)

Week 8 System Development Teams (2): Team Knowledge and Team Management

1. Majchrzak, A., More, P. H., & Faraj, S. (2012). Transcending knowledge differences in cross-functional teams. *Organization Science*, 23(4), 951–970.
2. Hsu, J., Chang, J., Klein, G., & Jiang, J. (2011). Exploring the impact of team mental models on information utilization and project performance in system development. *International Journal of Project Management*, 29(1), 1–12.
3. Kraut, R. E., & Streeter, L. A. (1995). Coordination in software development. *Communications of the ACM*, 38(3), 69–81.
4. Strode, D. E., Huff, S. L., Hope, B., & Link, S. (2012). Coordination in co-located agile software development projects. *Journal of Systems and Software*, 85(6), 1222–1238.

[Supplementary]

1. Huber, G. P., & Lewis, K. (2010). Cross-Understanding: Implications for Group Cognition and Performance. *The Academy of Management Review*, 35, 6–26.
2. Faraj, S., & Sproull, L. (2000). Coordinating expertise in software development teams. *Management Science*, 46(12), 1554–1568.
3. Schmidt, C., Kude, T., Heinzl, A., & Mithas, S. (2014). How Agile Practices Influence the Performance of Software Development Teams: The Role of Shared Mental Models and Backup. In the Proceedings of the International Conference on Information Systems (ICIS). Auckland, New Zealand.
4. Faraj, S., & Sambamurthy, V. (2006). Leadership of information systems development projects. *Engineering Management, IEEE Transactions on*, 53(2), 238–249.

Week 9 System Development Teams (3): The Impacts of ICTs

1. O’Leary, M. B., & Cummings, J. N. (2007). The spatial, temporal, and configurational characteristics of geographic dispersion in teams. *MIS Quarterly*, 31(3), 433–452.
2. Malhotra, A., Majchrzak, A., & Rosen, B. (2007). Leading virtual teams. *The Academy of Management Perspectives*, 21(1), 60–70.
3. Ziguers, I., & Munkvold, B. E. (2014). Collaboration technologies, tasks, and contexts. In *Human-Computer Interaction and Management Information Systems: Applications. Advances in Management Information Systems* (pp. 143–169).

Week 10 System Development Project: Commitment and Control

1. Keil, M., & Mähring, M. (2010). Is your project turning into a black hole. *California Management Review*, 53(1), 6–31.
2. Flynn, D., Pan, G., Keil, M., & Mähring, M. (2009). De-escalating IT projects: the DMM model. *Communications of the ACM*, 52(10), 131–134.
3. Kirsch, L. S. (1997). Portfolios of control modes and IS project management. *Information Systems Research*, 8(3), 215–239.

[Supplementary]

1. Maruping, L. M., Venkatesh, V., & Agarwal, R. (2009). A control theory perspective on agile methodology use and changing user requirements. *Information Systems Research*, 20(3), 377–399.
2. Napier, N. P., Keil, M., & Tan, F. B. (2009). IT project managers' construction of successful project management practice: A repertory grid investigation. *Information Systems Journal*, 19(3), 255–282.

Week 11 System Development Project: Performance Management, Success, and Failures

1. Aladwani, A. M. (2002). An Integrated Performance Model Information Systems Projects. *Journal of Management Information Systems*, 19(1), 185–210.
2. Gemino, A., Reich, B. H., & Sauer, C. (2007). A temporal model of information technology project performance. *Journal of Management Information Systems*, 24(3), 9–44.
3. Nelson, R. R. (2007). IT project management: Infamous failures, classic mistakes, and best practices. *MIS Quarterly Executive*, 6(2), 67–78.
4. [case discussion] Fitzgerald, G., & Russo, N. L. (2005). The turnaround of the London ambulance service computer-aided despatch system (LASCAD). *European Journal of Information Systems*, 14(3), 244–257.

[Supplementary]

1. Petter, S., DeLone, W., & McLean, E. (2008). Measuring information systems success: models, dimensions, measures, and interrelationships. *European Journal of Information Systems*, 17(3), 236–263.

Week 12 Global System Development

1. Dibbern, J., Winkler, J., & Heinzl, A. (2008). Explaining variations in client extra costs between software projects offshored to India. *MIS Quarterly*, 333–366.
2. Krishna, S., Sahay, S., & Walsham, G. (2004). Managing cross-cultural issues in global software outsourcing. *Communications of the ACM*, 47(4), 62–66.
3. Nagy, D., Yassin, A. M., & Bhattacharjee, A. (2010). Organizational adoption of open source software: barriers and remedies. *Communications of the ACM*, 53(3), 148–151.
4. Ven, K., & Verelst, J. (2011). An empirical investigation into the assimilation of open source server software. *Communications of the Association for Information Systems*, 28(9), 117–140.

[Supplementary]

1. Ramesh, B., Mohan, K., & Cao, L. (2012). Ambidexterity in Agile distributed development: An empirical investigation. *Information Systems Research*, 23(2), 323–339.
2. Crowston, K., Wei, K., Howison, J., & Wiggins, A. (2012). Free/Libre open-source software development: What we know and what we do not know. *ACM Computing Surveys (CSUR)*, 44(2), 1-35.