1. **Course Number and Course Title:**

   CMP 494-14 – Software Requirements Engineering

2. **Credit Hours:**

   3 – 0 – 3

3. **Prerequisites and/or Co-Requisites:**

   Prerequisite: CMP 305 (Data Structures and Algorithms)

4. **Name and Contact Information of Instructor:**

   Name: Dr. Raafat Aburukba
   Email: raburukba@aus.edu
   Office: ESB 2043
   Phone: 06 515 2953
   Office Hours: Posted on the office door and iLearn; also by appointment

5. **Course Description (Catalog Description):**

   Covers the principles, methods, and techniques used in software requirements engineering. Focuses on eliciting, analyzing, specifying, validating, and managing software requirements for complex systems. Includes a solid understanding of stakeholder identification and analysis. Emphasizes modeling and documentation of both functional and non-functional requirements, as well as the use of use cases and scenario analysis to capture system behavior. Explores requirements validation, verification, traceability, and management throughout the software development lifecycle.

6. **Textbook and other Supplemental Material:**

   Textbook:
   - Phillip Laplante, Mohamad Kassab, *Requirements Engineering for Software and Systems,* 4th ed., CRC Press, 2022.

   Other supplemental material:
   - Karl Wiegers, Joy Beatty, *Software Requirements,* 3rd ed., Microsoft Press, 2013.

7. **Course Learning Outcomes:**

   Upon completion of the course, students will be able to:
   1. Apply elicitation techniques to gather software requirements from stakeholders.
   2. Classify requirements as functional or non-functional using industry standards.
   3. Develop use cases, scenarios, and models to represent system requirements.
   4. Analyze requirements to ensure accuracy, completeness, and consistency.
   5. Prioritize requirements based on stakeholder needs and project constraints.
   6. Implement traceability and change management practices to handle evolving requirements.
   7. Produce a software requirements specification (SRS) document that meets quality standards.
   8. Communicate software requirements clearly in written and oral form.

8. **Teaching and Learning Methodologies:**

   Methods include lectures, problem-based learning, class discussions, and group work. Students learning is assessed via quizzes, exams, and projects.

9. Course Topics and Schedule:

Topic/Activity	Weeks
Introduction to Software Requirements Engineering	Week #1
Requirements Engineering Process Models	Week #2
Stakeholder identification and analysis	Week #3
Requirements elicitation techniques	Week #4
Advanced elicitation: ethnography, observation, prototyping	Week #5
Requirements analysis and negotiation	Week #6
Functional requirements modeling: use cases, user stories, and scenarios	Week #7
Non-functional requirements: quality attributes (performance, security, usability)	Week #8
Requirements specification techniques and best practices + Midterm Exam	Week #9
Requirements validation and verification	Week #10
Requirements management: versioning, traceability, and baselining	Week #11
Change management	Week #12
Requirements evolution	Week #13
Requirements in Agile vs. Plan-driven development	Week #14
Project presentations	Week #15
Final Exam	Week #16

10. Schedule of Laboratory and other Non-Lecture Sessions:

This course has no labs.

Project: The students will work in a team of 3 to deliver on an approved case study and deliver on the process of requirements engineering. The team will work on a Project Proposal, Stakeholder Analysis, Requirements Elicitation Report, System Requirements Specification, Use Case Model, Prototype, Requirements Validation Report, Change Management Plan, and Final Presentation.

11. Out-of-Class Assignments with Due Dates:

Assignment	Due Date (tentative)
Project: Project Proposal / Problem Statement	Week #2
Project: Stakeholder Analysis	Week #4
Project: Requirements Elicitation Report	Week #5
Project: System Requirements Specification (SRS)	Week #7
Project: Use Case Model / User Stories	Week #9
Project: Prototype	Week #10
Project: Requirements Validation Report	Week #12
Project: Change Management Plan	Week #13
Project: Presentation	Week #15

12. Student Evaluation:

Assessment	Weight	Due Date (tentative)
Quizzes	15 %	Biweekly
Project	25 %	cf. section 11
Midterm Exam	25 %	Week #9
Final Exam	35 %	Week #16

13. Assessment Instruments:

Assessment	Course Learning Outcomes
In-class Quizzes	O1 – O7

Project	O1 – O8
Midterm Exam	O1 – O3
Final Exam	O1 – O7

14. Contribution of Course to Program Outcomes:

BSCS Program Outcomes	**Emphasis in this course**	**Course Learning Outcomes**
(1) Analyze a complex computing problem and apply principles of computing and other relevant disciplines to identify solutions.	●	O1–O7
(2) Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.	○	O2–O7
(3) Communicate effectively in a variety of professional contexts.	◑	O1, O6, O7, O8
(4) Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.		
(5) Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.	○	O8
(6) Apply computer science theory and software development fundamentals to produce computing-based solutions.	◑	O3, O6, O7

Emphasis: ● High; ◑ Medium; ○ Low; Blank – Nothing Specific Expected

BSCoE Program Outcomes	**Emphasis in this course**	**Course Learning Outcomes**
(1) Identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics	◑	O1–O7
(2) Apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors	◑	O2 - O7
(3) Communicate effectively with a range of audiences	◑	O1, O6, O7
(4) Recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts		
(5) Function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives	○	O8
(6) Develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions		
(7) Acquire and apply new knowledge as needed, using appropriate learning strategies		

Emphasis: ● High; ◑ Medium; ○ Low; Blank – Nothing Specific Expected

15. Letter Grade Policy:

Total (T)	Letter Grade
$95 \leq T$	A
$90 \leq T < 95$	A-
$85 \leq T < 90$	B+
$80 \leq T < 85$	B
$75 \leq T < 80$	B-
$70 \leq T < 75$	C+
$65 \leq T < 70$	C
$60 \leq T < 65$	C-
$55 \leq T < 60$	D
$T < 55$	F