SECTION A: DEFINITIVE

1.	**General course information**		
1.1	School: Engineering and Digital Sciences	1.6	Credits (ECTS): 6
1.2	Course Title: Software Engineering	1.7	Course Code: CSCI 361
1.3	Pre-requisites: CSCI 152 Performance and Data Structures (C- and above)	1.8	Effective from: *Fall 2018*
1.4	Co-requisites: N/A		
1.5	Computer Science ☒ Core ☐ Elective Programs: *(in which the course is offered)*		

2.	**Course description (max.150 words)**

Software engineering is defined as the application of systematic and quantifiable approaches to the development and maintenance of software. A disciplined approach is essential given the highly complex software required by business and consumers. In this course, students will learn the fundamentals of software engineering including software processes and requirements engineering, project management, design, testing, and maintenance. Emphasis will be placed on agile software engineering methods and learning through hands-on coding assignments throughout the semester. Team based exercises will be used to teach project management skills. Coding will be done in Java and Javascript, and will include exposure to many of the standard tools from industry.

3.	**Summative assessment methods** (tick if applicable)**:**				
3.1	Examination	☒	3.5	Presentation	☐
3.2	Term paper	☐	3.6	Peer-assessment	☒
3.3	Project	☒	3.7	Essay	☒
3.4	Laboratory Practicum	☒	3.8	Other *(specify)*	_____

4.	**Course aims**

The aims of the course are:
1) to motivate an engineering approach to the design and development of software systems, familiarize students with industry best practices, and outline the relevant history
2) to help students cultivate a skillful approach to software development by providing practical exercises that illustrate common challenges

5.	**Course learning outcomes (CLOs)**
5.1	By the end of the course the student will be expected to be able:

1) to explain the software lifecycle and compare major process models such as waterfall and agile
2) to demonstrate effective software development practices through participation in a medium-sized team project
3) to show an awareness of and facility with a full range of software tools including version control, unit testing frameworks, and build tools

	4) to understand the use case model and to produce a software requirements specification 5) to explain principles of object-oriented design and show understanding of software design patterns 6) to describe the different levels of software testing (from unit to acceptance) and write test cases for non-trivial software modules 7) to demonstrate an awareness of key ethical concerns relating to modern software systems

5.2			

CLO ref #	Program Learning Outcome(s) to which CLO is linked	Graduate Attribute(s) to which CLO is linked
1,2,3,4,5,6	Identify and describe the significant issues, challenges, and milestones within the field; Assess technical problems and establish requirements for their solution; Design and implement substantive computer systems, in the form of devices or software; Identify the theoretical capabilities and practical limitations related to computing systems; Both function independently and serve effectively on a team to accomplish common goals;	Possess an in-depth and sophisticated understanding of their domain of study; Intellectually agile, curious, creative, and open-minded; Thoughtful decision-makers who know how to involve others;
7	Identify the social, ethical, legal, and security implications and responsibilities expected of a practicing professional in the field; Observe high levels of professional and personal conduct, as described by the university and the corresponding discipline professional societies;	Cultured and tolerant citizens of the world while being good citizens of their respective countries; Possess high personal integrity; Prepared to take a leading role in the development of their country.

2

| **SECTION B: NON-DEFINITIVE** |
| **Course Syllabus** |
| Details of teaching, learning and assessment |

6.	**Detailed course information**		
6.1	Academic Year: 2019-2020	6.3	Schedule (class days, time): Lecture: TBD Lab: TBD
6.2	Semester: Fall	6.4	Location (building, room): Lecture: TBD Lab: TBD
7.	**Course leader and teaching staff**		

Position	Name	Office #	Contact information	Office hours

Course Leader	Mark Sterling	#7E440	mark.sterling@nu.edu.kz	
Course Instructor(s)	Mark Sterling, Dinh-Mao Bui, Anh Tu Nguyen	#7E440	mark.sterling@nu.edu.kz mao.bui@nu.edu.kz tu.nguyen@nu.edu.kz	
Teaching Assistant(s)	TBD			

8.	**Course Outline**				
Session	**Date** (tentative)	**Topics and Assignments**		**Course Aims** (ref. # only, see item 4)	**CLOs**
Week 1		Introduction; Motivating software engineering; High profile software failures; Code of ethics		1, 2	1, 7
Week 2		Design Patterns		1, 2	2, 5
Week 3		Object-Oriented Development; Code Metrics		1, 2	2, 5
Week 4		Software Processes; Processes and Activities; Waterfall and Agile; Incremental and Iterative Development; Agile; Scrum		1, 2	1, 2
Week 5		Java Web Applications; Requirements		1, 2	1, 2
Week 6		Lean; Kanban; Front-End Development		1, 2	1, 2
Week 7		Quiz 1; Special Lecture		1, 2	2, 4
Week 8		Software Architecture; MVC		1, 2	3
Week 9		Refactoring		1, 2	3, 6
Week 10		Java Functional Programming and Types		1, 2	2, 3
Week 11		Concurrency		1, 2	1, 2
Week 12		Scripting and Shell Proficiency		1, 2	1, 2
Week 13		Quiz 2; Additional Topics		1, 2	1
Week 14		Special Topics; Course Review		1, 2	1 - 7

9.	**Learning and Teaching Methods**
1	Lecture-demonstration by teacher; Class projects; Homeworks.
2	Formal face-to-face lectures and office hours.
3	Team Project
4	Laboratory sessions to support lecture sections and provide with practical hands on experience
5	Occasional peer assessment of students' work

10.	**Summative Assessments (tentative)**			
#	**Activity**	**Date** (tentative)	**Weighting (%)**	**CLOs**
	Homework		20%	1-7
	Course Project		15%	1-7
	Quizzes (2)		25%	1-7
	Final Exam		30%	1-7
	Participation		10%	1-7

11.	**Grading**	
Letter Grade	**Percent range**	**Grade description** (where applicable)
A	95-100	
A-	90-94.9	
B+	85-89.9	
B	80-84.9	
B-	75-79.9	See Section 6 of "Academic Policies and Procedures for Undergraduate Programs" (available at https://registrar.nu.edu.kz/policies-and-procedures)
C+	70-74.9	
C	65-69.9	
C-	60-64.9	
D+	55-59.9	
D	50-54.9	
F	0-49.9	

12.	**Learning resources** (use a full citation and where the texts/materials can be accessed)	
E-resources, including, but not limited to: databases, animations, simulations, professional blogs, websites, other e-reference materials (e.g. video, audio, digests)	Readings are drawn from a large bibliography of material that we have compiled from scholarly sources. These readings are available through the library. The bibliography is part of a detailed set of lecture notes which are distributed via moodle. There are also presentation slides and code samples.	
E-textbooks	N/A	
Laboratory physical resources	Labs will be conducted in appropriate computer labs (e.g., 7-422, 7-522) with required software installed	
Special software programs	Eclipse for Java EE, Maven, Git	
Journals (inc. e-journals)	N/A	
Textbooks	Title: Applying UML and Patterns, Third Edition Author: Craig Larman Publisher: Prentice Hall PTR In addition to this text, there is also a detailed set of lecture notes.	

13.	**Course expectations**

Attendance
Missing classes and habitual tardiness will have a negative effect on your grade, both directly (through your attendance grade) and indirectly (by not benefitting from the in-class experience). Consistent with university policy, attendance below 80% will count significantly against your overall grade for the course and may result in failure.
You are also responsible for any announcements made during the class period, so be sure to ask your instructor, TAs, or classmates for any info that you may have missed if you did not attend.

Electronic Resources
Labs will be conducted in one of our hybrid computer labs, which are designed to accommodate the full range of course activities. The necessary programming tools are installed on the classroom lab computers, and are available online for free download to your own computer.
You are expected to check your Nazarbayev University e-mail on a daily basis for updates and announcements about the course.

Assignment Submission and Late Policy
Assignments must be submitted by the announced due date and time, as directed by the instructor. Some assignments may need to be submitted in the form of physical hard-copy in class, generally within the first five minutes of the start of the class period. Other assignments will need to be submitted digitally to Moodle. In case Moodle does not work, assignments need to be submitted by email to your instructor AND teaching assistant by the deadline date and time.

In general, there is no late policy; if you submit an assignment after it is due, you get zero points for your assignment. In cases of illness or family emergency, you must inform your instructor immediately if you believe you will not be able to submit your assignment on time. In such cases, an exception may be made at the discretion of your instructor.

Classroom Behavior
You are expected to act respectfully towards your fellow classmates, TAs, and instructors inside and outside of the classroom. We have a limited amount of time to cover a lot of material this semester, so you need to pay attention in class, and do your in-class work when it is as-signed. Talking on your phone, texting, chatting online, browsing Facebook or other social me-dia sites, and talking excessively with your neighbors about non-class related stuff are just a few examples of behavior that is not acceptable, and will negatively impact your grade.

14.	**Academic Integrity Statement**

Nazarbayev University and The School of Science and Technology have established high stand-ards for academic integrity, using an approach in which students are trained to produce original work according to professional standards, and to properly cite and reference the work of others when it is appropriate to do so.

The specific guidelines are published in the NU Student Handbook. In particular,

- The assignments in this class are designed to introduce important concepts and techniques, and enable you to explore the material independently so as to gain insight and comprehension of the subject. Doing the work is much more important than getting the right answer.

- The course is designed such that the new material presented each day builds on the skills developed in the preceding days; thus, any action that interferes with this process (missing class, skipping the assignment, copying) will seriously impede your progress.

- You are welcome—and encouraged—to talk through concepts and ideas with your fellow students and to study with them, but do not give or receive direct help from your classmates on a graded assignment.

- Assignments should be completed individually. If you distribute your work to others, even if you are not intending them to copy it, this is still considered academic misconduct.

- Even the appearance of cheating or inappropriate copying should be avoided.

- Students should be aware that the assignment submission process incorporates an automat-ed plagiarism detector.

- You may only get help on graded assignments from designated people—the instructors or TAs for the course. If you are struggling with an assignment, by all means, please seek help from them.

In the event that academic misconduct such as plagiarism or cheating is discovered, the student will receive no credit for the work, and the event reported to the Dean of your school. Egregious cases, or a second offense, can result in failure of the course and potential suspension or expulsion from the university.

When a student suspects that another student has violated the academic honesty policy, a re-port should be made to the appropriate faculty member.

15.	E-Learning

The primary resources are given in Section 12. For some special topics, we assign students to watch videos or listen to interviews that deal with thought leaders and influential organizations within the field.

16.	Approval and review		
Date of Approval:	Minutes #:	Committee:	
Date(s) of Approved Change:	Minutes #:	Committee:	