



Course Specification

— (Bachelor)

Course Title: **Code Generation**

Course Code: **SE1770**

Program: **Software Engineering**

Department: **Software Engineering**

College: **Faculty of Computer Science and Information Technology**

Institution: **Al-Baha University**

Version: **V1.0**

Last Revision Date: **24-April-2024**



Table of Contents

A. General information about the course:	3
B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods	4
C. Course Content	5
D. Students Assessment Activities	6
E. Learning Resources and Facilities	6
F. Assessment of Course Quality	7
G. Specification Approval	7



A. General information about the course:

1. Course Identification

1. Credit hours: (3)

3 Credit Hours (3, 0, 0) (Lecture, Lab, Tutorial)
(3 Contact Hours)

2. Course type

A. University College Department Track Others

B. Required Elective

3. Level/year at which this course is offered: (8th Level/3rd Year)

4. Course General Description:

This course provides a broad and in-depth coverage on Software Code Generation. The course discusses the subject of software code generation from the start of the notion to the current technologies used in the field of software industry. Nevertheless, the course pays a high attention on the theories and methodologies which code generations field is built on. In addition, the course 42 discusses the challenges and some applications of Software Code Generation along with CASE Tools of it.

5. Pre-requirements for this course (if any): None

6. Pre-requirements for this course (if any): None

7. Course Main Objective(s):

The main aim of this course is to teach the students the concepts of Software Code Generation. This includes discussion of the theories and methodologies which code generations field is built on. In addition, students will recognize the challenges of Software Code Generation and will be examine some CASE tools of Software Code Generation.

2. Teaching mode (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	Traditional classroom	33	100%
2	E-learning		
3	Hybrid <ul style="list-style-type: none"> • Traditional classroom • E-learning 		





No	Mode of Instruction	Contact Hours	Percentage
4	Distance learning		
5	Other (Lab)		

3. Contact Hours (based on the academic semester)

No	Activity	Contact Hours
1.	Lectures	33
2.	Laboratory/Studio	
3.	Field	
4.	Tutorial	
5.	Others (specify)	
Total		33

B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
1.0	Knowledge and understanding			
1.1	Recognize software code generation terminologies	K1	- Lectures - Exercise	Direct Assessment Tool Midterm Exam Final exam Indirect Assessment Tool Course Exit Survey
1.2	Identify the milestones in the history of software code generation	K2	- Lectures - Exercise	Direct Assessment Tool Midterm exam Final exam Indirect Assessment Tool Course Exit Survey
1.3	Illustrate the challenges of software code generation	K3	- Lectures - Exercise	Direct Assessment Tool Midterm exam Final exam Indirect Assessment Tool





Code	Course Learning Outcomes	Code of CLOs aligned with program	Teaching Strategies	Assessment Methods
				Course Exit Survey
1.4	Explain the theories and methodologies used software code generation	K4	- Lectures - Exercise	Direct Assessment Tool Midterm exam Final exam Indirect Assessment Tool Course Exit Survey
1.5	Interpret some potential applications of software code generation	K5	- Lectures - Exercise	Direct Assessment Tool Midterm exam Final exam Indirect Assessment Tool Course Exit Survey
2.0	Skills			
2.1	Examine some CASE tools of Software Code Generation	S1	- Lectures - Project	Direct Assessment Tool Project (rubric) Final exam Indirect Assessment Tool Course Exit Survey
3.0	Values, autonomy, and responsibility			
3.1	Recognize the ethical and professional use of Software Code Generation in the case of code ownership and copyrights	V1	-Small Groups	Direct Assessment Tool Project Presentation (rubric) Indirect Assessment Tool Course Exit Survey

C. Course Content

No	List of Topics (Lectures)	Contact Hours
1.	Introduction to Software Code Generation	4
2.	History of Software Code Generation	3
3.	Code Generation Theories	6
4.	Code Generation methodologies	6
5.	Code Generation Tools	6
6.	Code Generation challenges	4





7.	Applications of Code Generation	4
Total		33

D. Students Assessment Activities

No	Assessment Activities *	Assessment timing (in week no)	Percentage of Total Assessment Score
1.	Assignment	Weekly	10%
2.	Midterm Exam	6	20%
3.	Final Project and Presentation	11	20%
4.	Final Exam	12	50%

*Assessment Activities (i.e., Written test, oral test, oral presentation, group project, essay, etc.).

E. Learning Resources and Facilities

1. References and Learning Resources

Essential References	Herrington, Jack. Code generation in action. Manning Publications Co., 2003. Alexandre Peixoto de Queirós, Ricardo, Alberto Simoes, and Mario Teixeira Pinto, eds. Code Generation, Analysis Tools, and Testing for Quality. IGI Global, 2019.
Supportive References	Brambilla, Marco, Jordi Cabot, and Manuel Wimmer. "Model-driven software engineering in practice." Synthesis lectures on software engineering 3, no. 1 (2017): 1-207
Electronic Materials	<ul style="list-style-type: none"> • Access to the Saudi Digital Library (SDL). • Using the learning management system of the university – Rafid System (https://lms.bu.edu.sa/).
Other Learning Materials	<ul style="list-style-type: none"> • Nielsen Norman Group: https://www.nngroup.com/

2. Required Facilities and equipment

Items	Resources
facilities (Classrooms, laboratories, exhibition rooms, simulation rooms, etc.)	<ul style="list-style-type: none"> • A classroom or lecture hall with a whiteboard for 25 students.
Technology equipment (projector, smart board, software)	<ul style="list-style-type: none"> • A digital image projection system with a connection to a desktop computer and laptop computer. • High speed Internet connection





Items	Resources
Other equipment (depending on the nature of the specialty)	None

F. Assessment of Course Quality

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of teaching	<ul style="list-style-type: none"> • Students • Faculty • Peer Reviewers • Program Leader • Course Coordinator 	<ul style="list-style-type: none"> • Surveys (indirect). • Direct feedback from students. • Course evaluation by Peer Reviewers (indirect). • Class visit by Program Leader (indirect). • Comprehensive Course report (where we can find information about teaching difficulties and action plan)
Effectiveness of Students assessment	<ul style="list-style-type: none"> • Students • Faculty • Peer Reviewers • Program Leader • Exam Evaluation Committee • Course Coordinator 	<ul style="list-style-type: none"> • Surveys (indirect). • Direct feedback from students. • Course evaluation by Peer Reviewers (indirect). • Class visit by Program Leader (indirect) • Exam evaluation by the Exam Evaluation Committee (indirect)
Quality of learning resources	<ul style="list-style-type: none"> • Students • Faculty • Peer Reviewers • Course Coordinator 	<ul style="list-style-type: none"> • Surveys (indirect) • Course evaluation by Peer Reviewers (indirect). • Comprehensive Course report (where we can find information about difficulties and challenges about learning resources as well as consequences and action plan)
The extent to which CLOs have been achieved	<ul style="list-style-type: none"> • Faculty • Program Leader • Course Coordinator 	<ul style="list-style-type: none"> • Student Results (direct) • Comprehensive Course report (where we can find the CLO assessment results)
Other		

Assessors (Students, Faculty, Program Leaders, Peer Reviewers, Others (specify))

Assessment Methods (Direct, Indirect)

G. Specification Approval

COUNCIL /COMMITTEE	Curriculum Committee
---------------------------	-----------------------------





REFERENCE NO.

DATE

28 April 2024

