Department of Information and Communications Traffic
Chair of Telecommunication Traffic Theory
(in the establishment process)

UNIVERSITY OF ZAGREB
FACULTY OF TRANSPORT AND TRAFFIC SCIENCES

UNDERGRADUATE STUDY: TRANSPORT

SEMESTER (II)

Syllabus

Academic year 2025/2026

Course:	**Basics of Application Development**					
Head of course: Assoc. Prof. **Marko Matulin**, Ph. D.						
Co-lecturers: **Tomislav Erdelić**, Ph.D.						
Semester: **S**	Course code: **283269**	Lectures: **30**	Auditory exercises: **0**	Laboratory exercises: **8**	Seminars: **7**	ECTS credits: **3**
Group for lectures: **15 students**			Group for auditory and laboratory exercises: **15 students**			

The objective of the course:

- To educate students on how to plan, develop, and implement platform-independent applications, creating source code that meets and follows compatibility standards with modern browsers, devices, and operating systems.

Course enrolment requirements:

- To be eligible to enroll in the course, students need to have passed the course "Computer Science in Transportation Engineering" (exchange students must pass a similar computer science course to enroll).

Learning outcomes:

1. Devise an application solution for a given problem.
2. Connect databases with applications.
3. Design user interfaces compatible with different devices.
4. Develop various types of web applications.
5. Produce the implementation of the developed application using appropriate tools.

LECTURES, EXERCISES and SEMINARS

Week	Syllabus	Form of classes	Performed by	Lessons	Remark
1.	▪ Development methodologies for computer programs and documentation	L	Marko Matulin		
	▪ Applying an object-oriented approach to develop applications and modeling data structures	LE	Marko Matulin		
2.	▪ Architectures for developing application solutions	L	Marko Matulin		
	▪ Using different approaches to create a data model	LE	Marko Matulin		
3.	▪ Project setup and working in a programming environment	L	Marko Matulin		
	▪ Connecting models, views, and controllers	LE	Marko Matulin		
4.	▪ Description of project structure	L	Marko Matulin		
	▪ Linking the application to the database	LE	Marko Matulin		
5.	▪ Connecting the application to work with a database	L	Tomislav Erdelić		
	▪ Utilizing built-in libraries	LE	Tomislav Erdelić		

6.	▪ Frontend and backend data validation	L	Tomislav Erdelić		
	▪ Views and markup languages	LE	Tomislav Erdelić		
7.	▪ User role and access management	L	Marko Matulin		
	▪ Partial views	LE	Marko Matulin		
8.	▪ Creating user interfaces	L	Marko Matulin		
	▪ Validating data received from client devices and services	LE	Marko Matulin		
9.	▪ Dealing with exceptions	L	Marko Matulin		
	▪ Creating and documenting application models	S	Marko Matulin		
10.	▪ Application testing	L	Marko Matulin		
	▪ Performing basic operations for working with model objects	S	Marko Matulin		
11.	▪ Application performance management	L	Marko Matulin		
	▪ Working in programming languages and frameworks for web user interface development	S	Marko Matulin		

12.	▪ Deploying the application to a production environment	L	Marko Matulin		
	▪ Creating responsive user interfaces	S	Marko Matulin		
13.	▪ Tools for monitoring application operation and tracking user activity	L	Marko Matulin		
	▪ Working with cookies and session data	S	Marko Matulin		
14.	▪ Web application programming interfaces	L	Tomislav Erdelić		
	▪ Working with multimedia and integrating it into applications	S	Tomislav Erdelić		
15.	▪ Connecting applications to web services	L	Tomisav Erdelić		
	▪ Configuring the server to accept the application	S	Tomislav Erdelić		

L = Lectures; **AE** = Auditory Exercises; **LE** = Laboratory Exercises; S = Seminars

STUDENT OBLIGATIONS AND EXAMS

Conditions for obtaining signatures:
The course requires attendance to be taken throughout the semester, with students being obligated to attend at least 80% of the classes to be eligible to pass the course.

Written Examination:
Two mid-term exams are conducted during the semester, with students being able to earn a maximum of 50 points on each mid-term exam. To pass the written exam through mid-term exams,

a student must score at least 30 points on each mid-term exam. Students who fail to earn a minimum of 60 points on both mid-term exams must take a written exam. The written part of the exam carries a maximum of 100 points, and students must earn at least 60 points to pass the exam. The seminar task carries a maximum of 100 points, and a completed seminar is rated with a minimum of 60 points. The final grade is determined based on the achieved result on the mid-term exams or the written exam, as well as the points earned on the seminar, using the following range of points and final grades: 120-139 points = satisfactory, 140-159 = good, 160-179 = very good, 180-200 = excellent.

Oral examination:
There is no oral examination, apart from students' presentation of their seminar task to the instructor.

LITERATURE

a) Obligatory literature:

1. Matulin M., Erdelić T.: Lecture notes on Basics of Application Development (available at the faculty's online e-learning platform).
2. De Sanctis V. *ASP.NET Core 6 and Angular: Full-stack web development with ASP.NET 6 and Angular 13, 5th Edition.* Packt Publishing; 2022.

b) Recommended literature:

1. Kumar N. A Framework for User Interface and User Experience of Web-Based Application Paperback. Independent Author; 2023.

METHODOLOGY OF THE IMPLEMENTATION OF THE COURSE PLAN

1. LECTURES

Lectures cover the material presented in the authorized lectures (presentations and lecture notes) listed in the required literature and are mostly delivered using PowerPoint presentations. For some sections, video presentations are used. As part of the lectures, formative assessments of knowledge (through assignments) are conducted during the semester to provide feedback on how much of the material students have mastered and whether they are progressing, as well as to encourage and guide them in their learning, i.e., to give them feedback on their progress.

2. LABORATORY EXERCISES

Laboratory exercises are conducted in computer labs, using computers with various programming tools and platforms for developing applications and databases. A problem is presented to the students, which they solve during the exercises under the guidance of the instructor.

3. SEMINAR

At the beginning of the semester, students are given a seminar task to create a web application to solve a given problem. Throughout the semester, students work on the assigned application under the supervision of the instructor.

4. DOCUMENTATION

The student's attendance record is kept during the semester. Achievements of goals are recorded using the University information system ISVU. All tests are kept in the lecturer's file for one year.

Students may, at any time, request information about their current attendance rate as well as gain insight into their written exam (and colloquiums) in case the exam is rated unsatisfactory.

5. SCORING SYSTEM

Table 1 The scoring system for the monitoring of students and explained credit values in ECTS credits

no	Segment:		Required points to be achieved:		Remark:	ECTS credits
			Min.	Max.		
1.	Presence at the lectures		0	0	Students are obligated to attend at least 80% of the classes, i.e., 12 lectures, 6 laboratory exercises, and 6 seminar lectures but will not earn points.	**1**
2.	Presence at the exercises (auditory and laboratory)		0	0		**0.5**
3.	Seminar		60	100		**0.5**
4.	First colloquium	= Written exam	30	50		**0.5**
5.	Second colloquium		30	50		**0.5**
6.	Oral exam		/	/		/
Σ	*Overall points:*		120	200	*Overall ECTS credits:*	**3**

Table 2 - Explanation of the credit values in evaluations

POINTS:	Estimate based seminar paper and two colloquies (or written exam) - [3 ECTS]*	The final score [3 ECTS]
120-139	Sufficient (2)	Students can be awarded a maximum of 100 points on the written exam and another 100 points for seminar work.
140-159	Good (3)	
160-179	Very good (4)	
180-200	Excellent (5)	

*The student is evaluated based on the sum of the internal points earned through written exams and grading the seminar work. The student, to realize the condition "courses requirements achieved", must attend at minimum 80% of classes.

Information for students (scoring system, implementation plan, learning outcomes, syllabus, literature, consulting teachers, announcement of results of examinations or colloquium, and all other information):

- **https://moodle.srce.hr/2025-2026/ (https://merlin.srce.hr)**
- **https://www.isvu.hr/visokaucilista/en/podaci/135/nastavniprogram**

Student assistants: /