

BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM
VILLAMOSMÉRNÖKI ÉS INFORMATIKAI KAR

TANTÁRGYI ADATLAP

I. TANTÁRGYLEÍRÁS

1. ALAPADATOK

1.1. Tantárgy neve (magyarul, angolul)

Szoftverarchitektúrák

1.2. Azonosító (tantárgykód)

BMEVIAUMA21

1.3. A tantárgy típusa

-

1.4. Kurzustípusok és óraszámok (heti/féléves)

kurzustípus	óraszám (heti)	típus
előadás (elmélet)	3	-
gyakorlat	-	-
laboratóriumi gyakorlat	-	-

1.5. Tanulmányi teljesítményértékelés / értékelés típusa

félévközi érdemjegy

1.6. Kreditérték

5

1.7. Tantárgyfelelős

neve:	DR. Charaf Hassan
beosztása:	egyetemi tanár
elérhetősége:	charaf.hassan@vik.bme.hu

1.8. Tantárgyat gondozó szervezeti egység

Automatizálási és Alkalmazott Informatikai Tanszék (<https://www.aut.bme.hu>)

1.9. Tantárgyi adatlap elérhetősége

<https://www.aut.bme.hu/Course/VIAUMA21>

1.10. A tantárgy oktatásának nyelve

-

1.11. A tantárgy elsődleges mintatantervi jellege

1.12. A tantárgy előkövetelményei

erős előkövetelmény:	-
gyenge előkövetelmény:	-
párhuzamos előkövetelmény:	-
mérőföldkő előkövetelmény:	-
kizáró feltétel:	-
(Kizáró feltétel alkalmazása esetén az egyik tárgy teljesítése a másik tárgyból történő eltiltást jelenti.)	

2. CÉLOK ÉS TANULÁSI EREDMÉNYEK

2.1. Célkritizálás

The course provides an overview of cutting-edge software architectures used in applications, research, and development, discussing their role and significance in the development of information systems. Building on a review of current industry trends, it offers insights into future distributed and high-reliability system architectures and technologies expected to emerge. Previously introduced object-oriented, component-based, and service-oriented architectures are summarized in a synthesized form. The course emphasizes systematic software reusability and, in light of ongoing research in the field of software architecture, analyzes the challenges of designing loosely coupled systems as well as the importance of architectural patterns. Another key objective of the course is to organize and deliver up-to-date knowledge related to the above topics to the students. In enterprise systems development practice, multi-tier object-oriented platforms, microservices, and cloud technologies dominate. Practical experience demonstrates that these tools and technologies can effectively support application development; however, without adequate architectural expertise, numerous challenges may arise during implementation. Failures and unsuccessful development efforts often result from software engineers lacking sufficiently deep and broad architectural knowledge. In this regard, another primary aim of the course is to fully prepare students to perform professional development tasks for enterprise systems.

2.2. Tanulmányi eredmények

Kompetenciák:

A. Tudás

-

B. Képesség

-

C. Attitűd

-

D. Autonómia és felelősség

-

2.3. Oktatási módszertan

Követelmények: 1 db nagy zárthelyi 2 db kis zárthelyi 1 db házi feladat A tárgy teljesítésének feltétele a nagy zárthelyi elfogadható (legalább elégséges szintű) megoldása, és a 2 kis zh közül az egyik sikeres teljesítése, valamint a házi feladat megfelelő szintű teljesítése. A jegy kialakítása: ZH 40%, kis ZH-k 10%, házi feladat 50%. Pótlási lehetőségek: A zárthelyihez a TVSZ előírásai szerint mind a szorgalmi, mind a pótlási időszakban 1-1 pótlási lehetőséget biztosítunk.

2.4. Ajánlott irodalom

a) Tankönyv

-

b) Jegyzet

-

c) Online

1.

Patterns of Enterprise Application Architecture. Martin Fowler et al.

2.

Pattern-Oriented Software Architecture, Patterns for Concurrent and Networked Objects, Wiley press, vol.1 ,2.

2.5. Tantárgyleírás hatályossága

Kezdő időpont:

2026. március 21.

Záró időpont:

-

Tantárgyleírást elfogadó döntés:

-

II. TANTÁRGYKÖVETELMÉNYEK

3. TELJESÍTMÉNYÉRTÉKELES

3.1 Általános szabályok

Követelmények: 1 db nagy zárthelyi 2 db kis zárthelyi 1 db házi feladat A tárgy teljesítésének feltétele a nagy zárthelyi elfogadható (legalább elégséges szintű) megoldása, és a 2 kis zh közül az egyik sikeres teljesítése, valamint a házi feladat megfelelő szintű teljesítése. A jegy kialakítása: ZH 40%, kis ZH-k 10%, házi feladat 50%. Pótlási lehetőségek: A zárthelyihez a TVSZ előírásai szerint mind a szorgalmi, mind a pótlási időszakban 1-1 pótlási lehetőséget biztosítunk.

3.2 Értékelések

A. Évközi értékelések

B. Vizsgarészek

Vizsgarészek:

1. Írásbeli
-
2. Szóbeli
-
3. Gyakorlati
-
4. Évközi
-

3.3 Évközi értékelések részaránya

azonosító	részarány
-----------	-----------

3.4 Vizsgarészek részaránya

típus	részarány
Írásbeli	0 %
Szóbeli	0 %
Gyakorlati	0 %
Évközi	0 %

3.5 Érdemjegy megállapítása

-

A tanulási eredmények értékelése a fenti százalékos sávok alapján történik.

3.6 Jelenléti és részvételi követelmények

Elmélet: 70%

Gyakorlat: 70%

Labor: 70%

3.7 Javítás, pótlás, ismétlés

4. MUNKAIDŐ RÁFORDÍTÁS

4.1 Tevékenységek

Tevékenység	Óraszám
-------------	---------

Összesen

0

4.2 Hatályosság

Kezdő időpont:	2026. március 21.
Záró időpont:	-
Tantárgykövetelményt elfogadó döntés:	-

SUBJECT DATA SHEET

I. SUBJECT DESCRIPTION

1. BASIC DATA

1.1. Subject name

Software Architectures

1.2. Subject code

BMEVIAUMA21

1.3. Subject type

-

1.4. Course types and hours

course type	hours / semester	type
lecture	3	-
tutorial	-	-
laboratory	-	-

1.5. Assessment type

félévközi érdemjegy

1.6. Credit

5

1.7. Lecturer

name: DR. Charaf Hassan
position: egyetemi tanár
email: charaf.hassan@vik.bme.hu

1.8. Responsible department

Automatizálási és Alkalmazott Informatikai Tanszék (<https://www.aut.bme.hu>)

1.9. Subject website

<https://www.aut.bme.hu/Course/VIAUMA21>

1.10. Language of instruction

-

1.11. Primary curriculum

1.12. Prerequisites

Strong prerequisite: -
Weak prerequisite: -
Parallel prerequisite: -
Milestone prerequisite: -
Exclusion condition: -

(Exclusion condition means that completion of one subject excludes the other.)

2. OBJECTIVES AND LEARNING OUTCOMES

2.1. Objectives

The course provides an overview of cutting-edge software architectures used in applications, research, and development, discussing their role and significance in the development of information systems. Building on a review of current industry trends, it offers insights into future distributed and high-reliability system architectures and technologies expected to emerge. Previously introduced object-oriented, component-based, and service-oriented architectures are summarized in a synthesized form. The course emphasizes systematic software reusability and, in light of ongoing research in the field of software architecture, analyzes the challenges of designing loosely coupled systems as well as the importance of architectural patterns. Another key objective of the course is to organize and deliver up-to-date knowledge related to the above topics to the students. In enterprise systems development practice, multi-tier object-oriented platforms, microservices, and cloud technologies dominate. Practical experience demonstrates that these tools and technologies can effectively support application development; however, without adequate architectural expertise, numerous challenges may arise during implementation. Failures and unsuccessful development efforts often result from software engineers lacking sufficiently deep and broad architectural knowledge. In this regard, another primary aim of the course is to fully prepare students to perform professional development tasks for enterprise systems.

2.2. Learning outcomes

Competencies:

A. Knowledge

-

B. Skills

-

C. Attitudes

-

D. Autonomy and responsibility

-

2.3. Teaching methodology

Követelmények: 1 db nagy zárthelyi 2 db kis zárthelyi 1 db házi feladat A tárgy teljesítésének feltétele a nagy zárthelyi elfogadható (legalább elégséges szintű) megoldása, és a 2 kis zh közül az egyik sikeres teljesítése, valamint a házi feladat megfelelő szintű teljesítése. A jegy kialakítása: ZH 40%, kis ZH-k 10%, házi feladat 50%. Pótlási lehetőségek: A zárthelyihez a TVSZ előírásai szerint mind a szorgalmi, mind a pótlási időszakban 1-1 pótlási lehetőséget biztosítunk.

2.4. Recommended literature

a) Textbooks

-

b) Lecture notes

-

c) Online materials

1.

Patterns of Enterprise Application Architecture. Martin Fowler et al.

2.

Pattern-Oriented Software Architecture, Patterns for Concurrent and Networked Objects, Wiley press, vol.1 ,2.

2.5. Validity of the subject description

Start date:

2026. March 21.

End date: -

Subject Description Acceptance Decision: -

II. SUBJECT REQUIREMENTS

3. ASSESSMENT AND EVALUATION

3.1. General rules

Követelmények: 1 db nagy zárthelyi 2 db kis zárthelyi 1 db házi feladat A tárgy teljesítésének feltétele a nagy zárthelyi elfogadható (legalább elégséges szintű) megoldása, és a 2 kis zh közül az egyik sikeres teljesítése, valamint a házi feladat megfelelő szintű teljesítése. A jegy kialakítása: ZH 40%, kis ZH-k 10%, házi feladat 50%. Pótlási lehetőségek: A zárthelyihez a TVSZ előírásai szerint mind a szorgalmi, mind a pótlási időszakban 1-1 pótlási lehetőséget biztosítunk.

3.2. Assessments

A. Term evaluations

B. Exam components

Exam components:

1. Written
 -
2. Oral
 -
3. Practical
 -
4. Term result
 -

3.3. Weight of term evaluations

identifier	weight
------------	--------

3.4. Weight of exam components

type	weight
Written	0 %
Oral	0 %
Practical	0 %
Term result	0 %

3.5. Grading scale

-

Evaluation based on the table above.

3.6. Attendance requirements

Lecture: 70%

Tutorial: 70%

Laboratory: 70%

3.7. Retake and repeat

4. WORKLOAD

4.1. Activities

Activity	Hours
----------	-------

Total	0
--------------	----------

4.2. *Validity*

Start date:	2026. March 21.
End date:	-
Requirements Acceptance Decision:	-