UNIVERSITAS GADJAH MADA
FAKULTAS TEKNIK
DEPARTEMEN TEKNIK ELEKTRO DAN TEKNOLOGI INFORMASI

Rancangan Program Kegiatan Pembelajaran Semester (RPKPS)

1. Nama Matakuliah **: Arsitektur Perangkat Lunak**
2. Kode/SKS **: TKIT162207/2 SKS**
3. Prodi **:** S1 Teknologi Informasi
4. Status Matakuliah **: -**

5. Deskripsi Singkat Matakuliah:

Secara umum perangkat lunak dapat dimodelkan dengan 2 view utama, yaitu struktur dan behavior. Struktur berkaitan dengan arsitektur sedangkan behavior berkaitan dengan fungsionalitas atau bahkan dengan fitur-fitur suatu perangkat lunak. Secara khusus Matakuliah ini mempelajari tentang arsitektur perangkat lunak.

6. *Learning outcomes*:

1. TKIT162207-LO1 Students can argue the importance and role of software architecture in large-scale software systems
2. TKIT162207-LO2 Students can able to describe a software architecture using various documentation approaches and architectural description languages
3. TKIT162207-LO3 Student can use well-understood paradigms of software architecture for designing new systems
4. TKIT162207-LO4 Students can discuss and evaluate the current trends and technologies such as model-driven,service-oriented architectures, and cloud-based software.

9. Evaluasi yang direncanakan:

No	Bentuk Evaluasi	Pelaksanaan	Learning Outcome Dalam Persen (%)				Jumlah
			TKIT162207-LO1	TKIT162207-LO2	TKIT162207-LO3	TKIT162207-LO4	
1.	Tugas	Paruh Semester Awal	30	30	0	0	60
2.	UTS	Paruh Semester Awal	70	70	0	0	140
3.	Tugas	Paruh Semester Akhir	0	0	30	30	60
4.	UAS	Paruh Semester Akhir	0	0	70	70	140
Jumlah			**100**	**100**	**100**	**100**	

8. Materi Pembelajaran:

1. Pengenalan Arsitektur Perangkat Lunak
2. Software Architecture

3. Modeling and Notation
4. Quality Attributes
5. Component and Connector
6. Documenting Software Architecture (UML)
7. Evaluation
8. UTS
9. Middleware Architecture and Technologies
10. Software Product Lines
11. Model-Driven Architecture
12. Service-Oriented Architecture
13. Aspect Oriented Architecture
14. Architecture in the Cloud
15. UAS

10. Bahan, sumber informasi, dan referensi:

1. Len, Bass, Clements Paul, and Kazman Rick. "Software architecture in practice." *Boston, Massachusetts Addison* (2003).
2. Gorton, Ian. *Essential software architecture*. Springer Science & Business Media, 2006.

11. Rencana Kegiatan Pembelajaran Mingguan (RKBM):

Minggu Ke	Topik (Pokok Bahasan)	Metode Pembelajaran
(1)	(2)	(3)
1.	**1 What Is Software Architecture?** 1.1 What Software Architecture Is and What It Isn't 1.2 Architectural Structures and Views 1.3 Architectural Patterns 1.4 What Makes a "Good" Architecture?	Presentasi Diskusi
2.	**2 Why Is Software Architecture Important?** 2.1 Inhibiting or Enabling a System's Quality Attributes 2.2 Reasoning About and Managing Change 2.3 Predicting System Qualities 2.4 Enhancing Communication among Stakeholders 2.5 Carrying Early Design Decisions 2.6 Defining Constraints on an Implementation 2.7 Influencing the Organizational Structure 2.8 Enabling Evolutionary Prototyping	Presentasi Diskusi
3.	**Quality Attribute** • Availability • Interoperability • Modifiability • Testability • Usability	Presentasi Diskusi
4.	**Visualizing Software Architecture** • Textual • Graphic • Hybrid	Presentasi Diskusi
5.	**5. UML and Documenting Architecture**	• Presentasi • Diskusi

Minggu Ke	Topik (Pokok Bahasan)	Metode Pembelajaran
(1)	(2)	(3)
	• UML Structural component • UML Behavioral component • UML 2.0 • Architecture Document Template	• Tugas perancangan menggunakan UML
6.	**6. Design Pattern** • Creational pattern • Structural pattern • Behavioral pattern	Presentasi Diskusi
7.	Presentasi Mahasiswa 1. Penerapan arsitektur design pattern	Presentasi mahasiswa Diskusi
8.	Ujian Tengah Semester	
9.	**Middleware Architectures and Technologies** 4.1 Introduction 4.2 Middleware Technology Classification 4.3 Distributed Objects 4.4 Message-Oriented Middleware 4.4.1 MOM Basics 4.4.2 Exploiting MOM Advanced Features 4.4.3 Publish–Subscribe 4.5 Application Servers 4.5.1 Enterprise JavaBeans 4.5.2 EJB Component Model 4.5.3 Stateless Session Bean Programming Example 4.5.4 Message-Driven Bean Programming Example 4.5.5 Responsibilities of the EJB Container	Presentasi Diskusi
10.	**Service-Oriented Architectures and Technologies** 5.1 Background 5.2 Service-Oriented Systems 5.2.1 Boundaries Are Explicit 5.2.2 Services Are Autonomous 5.2.3 Share Schemas and Contracts, Not Implementations 5.2.4 Service Compatibility Is Based on Policy 5.3 Web Services 5.4 SOAP and Messaging 5.5 UDDI, WSDL, and Metadata 5.6 Security, Transactions, and Reliability 5.7 RESTful Web Services	Presentasi Diskusi Referensi: Ch 12 textbook
11.	**Model Driven Architecture** 1. Kaitan MDA dengan OOP 2. Pengembangan software menggunakan konsep MDA 3. State of the art practice and tool 4. MDA and Software Architecture 5. MDA and Nonfunctional Requirements 6. Model Transformation and Software Architecture	Presentasi Diskusi Referensi: Ch 14 textbook

Minggu Ke	Topik (Pokok Bahasan)	Metode Pembelajaran
(1)	(2)	(3)
12.	**Dependency Injection Architecture** • Introduction • Benefit DI • Bets practice	Presentasi Diskusi
13.	**13 Aspect Oriented Architectures** • Aspects for ICDE Development • Introduction to Aspect-Oriented Programming • Aspect-Oriented Architectures • Architectural Aspects and Middleware • State-of-the-Art • Aspect Oriented Modeling in UML • AOP Tools • Annotations and AOP • Performance Monitoring of ICDE with AspectWerkz	Presentasi Diskusi Ref : Ch 13
14.	**Architecture in the Cloud** • Basic Cloud Definitions • Service Models and Deployment Options • Economic justification • Base Mechanisms • Sample Technologies • Architecting in a Cloud Environment	Presentasi Diskusi Referensi: Ch 26 Textbook
15.	**Microservice Architecture** • Introduction • Benefit • Best Practice	Presentasi Diskusi
16.	Ujian Akhir Semester	