

## BBM 382-Software Engineering Final Exam

31.May.2017 09:30

Duration: 100 minutes (Closed book & notes)

Instructor: Dr. Ayça Tarhan

Student ID: .....  Student Name: .....  Signature: .....  Section: <input type="checkbox"/> #1 (Dr. Vahid Garousi) <input type="checkbox"/> #2 (Dr. Ayça Tarhan)	Question	Points	Your mark
	1	16	
	2	10	
	3	12	
	4	16	
	5	10	
	6	16	
	7	20	
	<b>Total</b>	<b>100</b>	

**Question-1)** Describe basics of Agile software development methodology by listing four elements that it **values more** in comparison to the elements valued in Plan-Driven methodology.

Agile Methodology (values these factors)	(over these in) Plan-Driven Methodology
1) Individuals and interactions	processes and tools
2) Working software	comprehensive documentation
3) Customer collaboration	contract negotiation
4) Responding to change	following a plan

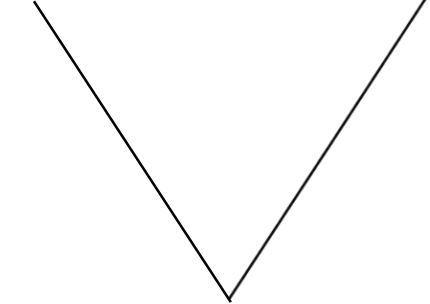
**Question-2)** Assume you are **leading an in-house software development team** that develops software for use by employees working in various departments (e.g. Accounting, Sales, HR, etc.) in a **government-based organization**. Would you use Waterfall or Iterative software development methodology? Explain the rationale of your selection in comparison to the one that you discarded.

Your selection (Waterfall / Iterative ?)	Rationale of your selection (Why did you choose it?)	Rationale of discard for the other (Why did you discard the other?)
Iterative	<p><i>(Any two answers will be accepted)</i></p> <ul style="list-style-type: none"> <li>- Allows interaction with customer during development</li> <li>- Customer requirements can be prioritized and be developed by increments</li> <li>- Any change requests can always be included in the next iterations</li> <li>- Customer satisfaction is more important than obeying to contract requirements</li> </ul>	<p><i>(Any two answers will be accepted)</i></p> <ul style="list-style-type: none"> <li>- Customers typically do not interact with development team during development</li> <li>- Requires complete specification of requirements before development</li> <li>- Does not welcome changes after requirements stage</li> <li>- Typically requires a contract, which is not the case with in-house development</li> </ul>

**Question-3)** Describe four basic activities of **Software Risk Management** with their outputs. Also give two examples to typical risks that can be observed in a software project.

Activity	Description	Output
1) Risk Identification	You identify possible project, product, and business risks.	List of Potential Risks
2) Risk Analysis	You assess the likelihood and consequences of these risks.	Prioritized Risk List
3) Risk Planning	You make plans to address the risk, either by avoiding it or minimizing its effects on the project.	Risk Avoidance and Contingency Plans
4) Risk Monitoring	You regularly assess the risk and your plans for risk mitigation and revise these when you learn more about the risk.	Risk Assessment
Examples to typical software risks	<p><i>(Any other valid answer will be accepted – you may refer to Textbook pg. 599, Figure 22.3)</i></p> <p>1) (Technology) Reusable software components contain defects that mean they cannot be reused as planned.</p> <p>2) (People) It is impossible to recruit staff with the skills required.</p>	

**Question-4)** List types of reviews (with names of development phases) and **levels of testing** that can typically be performed while applying **V-development model** given below.

Reviews		Tests
Requirements review		System testing
Architectural (technical) review		Integration testing
Detailed design (technical) review or inspection		Component testing
Code review or inspection		Unit testing

**Question-5)** Describe four basic stages of **Software Change Management**. Also define the concepts of **version** and **baseline** in the context of Software Configuration Management.

1) Change request	a 'customer' completes and submits a change request describing the change required to the system
2) Change assessment	The impact of the change on the rest of the system must be checked. A separate group decides if it is cost-effective from a business perspective to make the change.
3) Change implementation	Accepted changes are passed back to the development group for implementation.
4) Change validation	Implemented changes are validated by software testing (as any other requirements implementation), and QA people approves the implementation of the change.
<b>Version:</b>	An instance of a configuration item that differs, in some way, from other instances of that item.
<b>Baseline:</b>	A baseline is a collection of component versions that make up a system. Baselines are controlled, which means that the versions of the components making up the system cannot be changed.

**Question-6) Map the most suitable software applications to software architecture types below. Then describe each architecture type by a sentence and explain your rationale for each mapping.**

Software Application	(Map)	Architecture Type	Description of Architecture Type & Mapping Rationale
MS Windows Operating System		Model-View-Controller	Separates presentation and interaction from the system data. Used when there are multiple ways to view and interact with data. / Students, supervisors, and administrators interact in different ways with data in different times (registration period vs. in-term)
Spotify Web Player		Layered	Organizes the system into layers with related functionality associated with each layer. Used when building new facilities on top of existing systems. / OS architecture (kernel and shell, each with its sub-layers) fits well with layered architecture.
HU Course Registration System		Repository	All data in a system is managed in a central repository that is accessible to all system components. You should use this pattern when you have a system in which large volumes of information are generated that has to be stored for a long time. / Development data are various and large in volumes and need storage for a long time.
Github		Client-Server	The functionality of the system is organized into services, with each service delivered from a separate server. Used when data in a shared database has to be accessed from a range of locations. / Music database and its features are accessible by any client user at any geographic location via replicated servers.

**Question-7) List views in 4+1 view model of software architecture below and explain each view with a sentence. For each view, also list the most suitable UML diagrams used for representation.**

Name of View	Explanation of View	UML Diagram(s) used
1) Logical view	It shows the key abstractions in the system as objects or object classes.	Class diagram, or object diagram, or composite structure diagram
2) Process view	It shows how, at run-time, the system is composed of interacting processes.	Sequence diagram, or communication diagram
3) Development view	It shows the breakdown of the software into components that are implemented by a single developer or development team.	Package diagram, or component diagram
4) Physical view	It shows the system hardware and how software components are distributed across the processors in the system.	Deployment diagram
(+1) Scenario View	It shows the interaction between users and the system for operational scenarios.	Use case diagram, or activity diagram

