**Pjesa 1: Pyetje me alternativa**

**Pjesa 2: Pyetje me përgjigje të shkurtra**

**Kuptimi i Arkitekturës së Softuerit**

1. **Çfarë është Arkitektura e Softuerit?**  
   Arkitektura e softuerit përfaqëson strukturën themelore të një sistemi softuerik, përfshirë komponentët e tij, marrëdhëniet midis tyre dhe parimet që udhëheqin dizajnin dhe evolucionin e tij.
2. **Çfarë bën një Arkitekt i Softuerit?**  
   Ai planifikon, harton dhe mbikëqyr strukturën e përgjithshme të sistemit, siguron që arkitektura të plotësojë kërkesat funksionale dhe jo-funksionale, dhe ndihmon ekipin zhvillues në marrjen e vendimeve teknike.
3. **Pse është e rëndësishme Arkitektura e Softuerit?**  
   Sepse ndikon në performancën, shkallëzueshmërinë, mirëmbajtjen, sigurinë dhe kostot e zhvillimit dhe mirëmbajtjes së një sistemi.
4. **Cilat janë katër pamjet (model views) e arkitekturës?**

* Pamja Logjike (Logic View)
* Pamja e Zbatimit (Deployment)
* Pamja e Procesit (Process View)
* Pamja Fizike (Development)

1. **Cilat janë tre zonat kryesore të kërkesave jo-funksionale?**

* Cilësitë e softuerit (Quality attributes)
* Kufizimet teknologjike
* Kërkesat e biznesit

**Stilet dhe Modelet Arkitekturore**

1. **Shpjego termat “Stil Arkitekturor” dhe “Model Arkitekturor”:**

* *Stili* është një grup karakteristikash që përshkruan një mënyrë të përgjithshme për ndërtimin e sistemeve.
* *Modeli* është një zbatim specifik i një stili për të zgjidhur një problem të caktuar.

1. **Listo tri kategori të stileve:**

* Stilet e Modulimit (Module styles)
* Stilet e Komunikimit (Component-and-Connector)
* Stilet e Zbatimit (Deployment styles)

1. **Stile të zakonshme të modulimit dhe përshkrimi i njërit:**

* Layered, Pipe-and-Filter, Client-Server  
  P.sh. *Layered* – Sistemi ndahet në shtresa që secila ka një përgjegjësi të veçantë dhe ndërvepron vetëm me shtresat fqinje.

1. **Çfarë janë Komponentët dhe Lidhëset (Connectors)?**

* *Komponentët* janë njësi të pavarura që kryejnë funksione specifike.
* *Connectors* janë mekanizma që mundësojnë komunikimin ndërmjet komponentëve.

**Cilësitë e Softuerit**

1. **Çfarë janë kërkesat për cilësitë e softuerit?**  
   Janë kërkesa që nuk lidhen me funksionalitetin, por me mënyrën se si softueri funksionon, si performanca, siguria, shkallëzueshmëria etj.
2. **Cilat janë disa cilësi të përgjithshme të softuerit (“-ilitë”)?**

* Reliability, Availability, Scalability, Maintainability, Usability, Security

1. **Shpjegim i shkurtër i Authentication dhe Authorization:**

* *Authentication* – Verifikimi i identitetit të përdoruesit.
* *Authorization* – Përcaktimi i asaj që përdoruesi ka leje të bëjë.

1. **Shpjegim i “Performancës” dhe shembull nga projekti juaj:**  
   Performanca është aftësia e sistemit për të reaguar shpejt dhe efikasisht.  
   ***Shembull:*** *Në projektin tim Food Ordering App, zgjedhja e produkteve per te shtuar ne shport dhe berja e porosive duhet të ndodhin brenda 1-2 sekondave për të ofruar një përvojë të shpejtë për përdoruesit.*

**Procesi i Arkitekturës së Softuerit**

1. **Tre hapat e procesit të dizajnit arkitekturor (me diagram):**
   * Përcaktimi i kërkesave
   * Dizajni
   * Validimi
2. **Input-et dhe Output-et që përcaktojnë kërkesat arkitekturore:**  
   Input: Kërkesat funksionale, jo-funksionale, kufizimet  
   Output: Dokumenti i kërkesave të arkitekturës.
3. **Input dhe output i dizajnit të arkitekturës:**  
   Input: Kërkesat arkitekturore  
   Output: Dizajni arkitekturor, modelet, pamjet e arkitekturës dhe dokumenti i dizajnit
4. **Cili është qëllimi i fazës së Validimit?**  
   Të vërtetojë që arkitektura është e përshtatshme dhe funksionale për qëllimin e sistemit.

**Arkitektura në Projekte AGILE**

1. **Parime nga Agile Manifesto (minimumi 4):**

– Kënaqësia e klientit me dorëzim të shpejtë  
– Mirëpritja e ndryshimeve edhe në faza të vona  
– Bashkëpunimi i përditshëm mes biznesit dhe zhvilluesve  
– Softueri funksional si matës i progresit

1. **Metoda AGILE (minimumi 3):**

Scrum, Kanban, XP, Lean, FDD, DSDM, AUP, ASD, Crystal Clear

1. **Tre rolet kryesore në SCRUM:**

* *Product Owner* – Përfaqëson klientin dhe Menaxhon kërkesat dhe prioritetet
* *Scrum Master* – Siguron ndjekjen e procesit dhe heq pengesat.
* *Developers* – Implementojnë kërkesat, zhvillojn produktin

1. **Karakteristika të AGILE në SDLC:**

* Iteracione të shkurtra
* Përshtatshmëri ndaj ndryshimeve
* Përfshirje e vazhdueshme e klientit
* Dorëzimi i vazhdueshëm i softuerit funksional

**Arkitektura Monolitike & Mikroshërbime, Docker**

1. **Krahasim mes konteinerëve dhe makinave virtuale:**  
   Konteinerët ndajnë kernelin dhe janë më të lehtë. VM-të janë më të rënda dhe përdorin më shumë burime.
2. **Përfitimet e arkitekturës monolitike:**

* Thjeshtësi në ndërtim dhe testim
* Performancë më e lartë
* Zbatim më i lehtë për aplikacione të vogla

1. **Përfitimet e mikroshërbimeve:**

* Shkallëzueshmëri më e mirë
* Çdo shërbim mund të zhvillohet dhe vendoset në mënyrë të pavarur
* Përshtatshmëri më e madhe me teknologji të ndryshme

1. **Tre avantazhe të përdorimit të Docker mbi VM tradicionale:**

* Fillimi i shpejtë
* Migrim i lehtë ndërmjet mjediseve
* Përdorim më i ulët i burimeve
* Konsistencë në ambient
* Mbështet CI/CD dhe DevOps

1. **Përshkruaj arkitekturën e Docker:**

Docker përdor imazhe për të krijuar konteinerë, që ruhen në registries si Docker Hub. Siguron izolim dhe efikasitet*.*

**UML dhe Dokumentimi i Arkitekturës së Softuerit**

1. **Çfarë është UML?**  
   Unified Modeling Language - Gjuhë standarde vizuale për të modeluar, dokumentuar dhe analizuar sisteme softuerike.
2. **Çfarë përfaqësojnë diagramet strukturore?**  
   Përfaqësojnë arkitekturën statike (Class, Component, Deployment, etj.).
3. **Çfarë përfaqësojnë diagramet e sjelljes?**  
   Përshkruajnë mënyrën se si sistemi sillet në kohë reale (p.sh., use-case, sequence).
4. **Shembull i diagramës UML Sequence:**

Përfaqësojnë ndërveprimet dhe ndryshimet në sistem (Sequence, Use Case, Activity, etj.).  
  
(Në versionin PDF apo në letër vizato një diagram që përfshin një përdorues, një ndërfaqe, dhe një bazë të dhënash me thirrje të funksioneve në radhë.)

1. **Tre kategori kryesore të notacioneve për dokumentimin e arkitekturës:**

– Structural  
– Behavioral  
– Textual

1. **Tre hapat për zgjedhjen e pamjeve/view:**
   * Krijo tabelën stakeholder/view
   * Kombino pamjet
   * Prioritizo dhe organizo

**Dizajnimi i Arkitekturës**

1. **Cila metodë përdoret për të përmbushur kërkesat cilësore dhe funksionale?**  
   Attribute-Driven Design (ADD)
2. **Pesë hapat e metodës ADD:**
3. Dizajni i të dhënave
4. Struktura arkitekturore
5. Analiza e stileve
6. Përzgjedhja e stilit
7. Elaborimi i arkitekturës
8. **Tre strategjitë kyçe të dizajnit:**

* Modularizimi
* Abstraksioni
* Përdorimi i stileve dhe modeleve ekzistuese

– Decomposition  
– Fokusimi në kërkesat arkitekturore  
– Generate and test

**Vlerësimi i Arkitekturës**

1. **Llojet e vlerësimeve të arkitekturës:**

* Vlerësimi formal
* Vlerësimi i bazuar në skenarë
* Rishikimi teknik

– Teknik: ATAM  
– Ekonomik: CBAM

1. **Cila metodë duhet përdorur për të vlerësuar një arkitekturë softuerike?**  
   ATAM – Architecture Tradeoff Analysis Method
2. **Cila metodë është përdorur për një dekadë për vlerësimin e arkitekturës?**  
   ATAM
3. **Cilët janë tre grupet që marrin pjesë në ATAM:**

* Palët e interesit (stakeholders)
* Vlerësuesit (analistët)
* Ekipi zhvillues