**lecture 13: تحليل معماري نرم افزار**

**تحليل معماري (Architectural Analysis)**

تحليل معماري فعاليتي است كه در آن با استفاده از مدل هاي معماري سيستم، به خصوصيات مهم سيستم پي مي بريد.

در تحليل معماري مهم است كه بدانيم:

1. چه سوالاتي را بپرسيم.
2. چرا اين سوالات را مي پرسيم.
3. چگونه اين سوالات را بپرسيم.
4. چگونه مطمئن شويم كه اين سوالات مي توانند پاسخ داده شوند.

**نكته:** تحليل معماري بعد از مدلسازي آمده است و مدل طراحي شده است.

**تحليل مدل هاي معماري غيرصوري**

**مزايا:**

* به معمار كمك مي كند كه از مشتري هاي سيستم توضيح روشني بگيرد.
* به مديران كمك مي كند كه از حوزه پروژه مطمئن شوند.

**معايب:**

* براي توسعه دهندگان (developers) مفيد نيست.(براي تمامي stakeholder ها مفيد نمي باشد.)
* داراي ابهام مي باشد.

**تحليل مدل هاي معماري صوري**

**مزايا:**

* به معماران براي تعيين كامپوننت ها كمك مي كند.
* به توسعه دهندگان براي تصميمات سطح پياده سازي كمك مي كند.
* به توليد خودكار كد كمك مي كند.

**معايب:**

* سختي براي افرادي كه در حوزه نرم افزار كار مي كنند.
* محدوديت فضاي انفجار (state space explosion)

**نكته:** مزيت روش هاي formal بر informal اين است كه در بخش تحليل دقيق تر مي باشند.

**نكته:** در روش هاي informal، مبهم بودن باعث مي شود كه توليد خودكار متوقف شده و همه كارها manual انجام شود.

**نكات مهم در تحليل معماري**

* اهداف تحليل (Goals of analysis)
* حوزه تحليل (Scope of Analysis)
* مفاهيم مورد تحليل
* سطح صوري بودن (Level of Formality)
* نوع تحليل (Type of analysis)
* سطح اتوماسيون (Level of automation)
* stakeholder هاي سيستم

**اهداف تحليل معماري**

1. كامل بودن (**C**ompleteness)
2. هم خوان بودن (**C**onsistency)
3. سازگاري (**C**ompatibility)
4. صحت (**C**orrectness)

**كامل بودن (Completeness)**

‍Completeness هم يك هدف بيروني (external) و هم يك هدف دروني (internal) است.

**هدف بيروني (external):**

تمام نيازمندي هاي سيستم را پوشانده باشد.

**هدف دروني (internal):**

اجزا و عناصري كه در كنار هم قرار گرفته اند تا نرم افزار را تشكيل دهند، كامل ترين و دقيق ترين اجزا بوده اند.

**هم خوان بودن (Consistency)**

يكي از ويژگي هاي دروني (internal) يك مدل معماري است.

تضمين مي كند كه المان هاي مختلف نرم افزار با هم در تضاد نيستند.

**ابعاد consistency:**

* نام (Name)
* واسط (Interface)
* رفتار (Behavior)
* تعامل (Interaction)
* پالايش (Refinement)

**هم خواني نام (Name Consistency)**

كامپوننت ها و كانكتورهاي هم نام نداشته باشيم.

سرويس هاي كامپوننت ها(متدها) هم نام نباشند.

**هم خواني واسط (Interface Consistency)**

واسط هاي هم نام نداشته باشيم.

**هم خواني رفتاري (Behavioral Consistency)**

رفتار هر تابع با نامي كه به آن اختصاص داده شده است سازگار باشد.

**هم خواني تعامل (Interaction Consistency)**

بر اساس پروتكل موجود، تعامل بين كامپوننت ها برقرار شود. مثلا اگر در پروتكل، ترتيب خاصي در تعامل بين كامپوننت ها وجود دارد، اين ترتيب رعايت شود.

**هم خواني پالايش (Refinement Consistency):**

تمام خصوصيات مدل سطح بالا بايد در مدل سطح پايين حفظ شده باشد و همچنين هر خاصيتي در مدل سطح پايين، نمودي از يك خاصيت در مدل سطح بالا باشد.

**سازگاري (Compatibility)**

يكي از ويژگي هاي بيروني (external) يك مدل معماري است.

تضمين مي كند كه مدل معماري با style، metal-model يا استانداردي كه از آن پيروي كرده ايد، سازگار است.

**نكته:** Consistency و Compatibility يكسان هستند با اين تفاوت كه Consistency يك ويژگي دروني است و Compatibility يك ويژگي بيروني مي باشد.

**صحت (Correctness)**

يكي از ويژگي هاي بيروني (external) يك مدل معماري است.

تضمين مي كند كه:

1. نيازمندي ها را به درستي Specification كرده ايم.
2. سيستم به درستي درك شده و نيازمندي ها به درستي شناخته شده اند.

**Specification:** فرآيند شناخت و طراحي حاصل از اين شناخت.

**حوزه تحليل**

1. **سطح كامپوننت و كانكتور**

هر كامپوننت و كانكتور مورد تحليل قرار گيرد.

1. **سطح زير سيستم و سيستم**

در قالب زير سيستم و تعاملات ميان آنها، تحليل انجام شود.

1. **تبادل داده ها(Data Exchange) در يك سيستم يا زيرسيستم**

ساختارهاي داده اي طراحي شده در زيرسيستم هاي مختلف، مورد تحليل قرار گيرند.

1. **سطوح مختلف انتزاع**

بعضي از آناليزها را بايد در سطوح بالاي انتزاع انجام دهيم.

هر چه سطح انتزاع بالاتر باشد، حوزه ويژگي هاي قابل بررسي وسيع تر است.

1. **مقايسه دو يا چند معماري از طريق:**

* پردازش (Processing)
* داده ها (Data)
* تعامل (Interaction)
* پيكربندي (Configuration)
* ويژگي هاي غيركاركردي (Non-functional properties)

**مفاهيمي كه بايد مورد تحليل قرار گيرند**

1. خصوصيات ساختاري (Structural characteristics)
2. خصوصيات رفتاري (Behavioral characteristics)
3. خصوصيات تعاملي (Interaction characteristics)
4. خصوصيات غيركاركردي (Non-functional characteristic)

**سطح formality**

1. مدل هاي غير صوري (Informal models)
2. مدل هاي نيمه صوري (Semi-formal models)
3. مدل هاي صوري (formal models)

**نوع تحليل**

1. **تحليل ايستا**

بيشتر مربوط به خصوصيات ساختاري است.

1. **تحليل پويا**

بيشتر مربوط به خصوصيات اجرايي است.

1. **تحليل Scenario-driven**

يك سناريو از كامپوننت ها و تعاملات ميان آنها را ايجاد مي كند.

براي سيستم هاي بزرگ استفاده مي شود.

مي تواند هم شامل تحليل ايستا و هم تحليل پويا باشد.

**سطح اتوماسيون**

1. **دستي (Manual)**

وقتي از زبان هاي غيرصوري استفاده مي كنيم.

1. **تا حدي خودكار (Partially automated)**

وقتي از زبان هايي مثل UML استفاده مي كنيم.

1. **كاملا خودكار (Fully automated)**

قابليت تحليل به طور خودكار را دارند.

**stakeholder هاي تحليل**

1. معماران (Architects)
2. توسعه دهندگان (Developers)
3. مديران (Managers)
4. مشتريان (Customers)
5. فروشندگان (Vendors)