TDT4240 Software Architecture

ATAM Evaluation

Marius Greve Hagen
David Rozas Domingo
Maria Fernandez-Rodriguez
Jarle Lindseth

Introduction

The following document shows the analysis performed to the XNA video game project of the group 18. This project is focus on testability.
The analysis has been performed by the group 17, whose members are María Fernandez, Jarle Lindseth, Marius Greve and David Rozas.

Attribute utility tree

In order to build the utility tree, we examined carefully the scenarios the architecture designers had provided.

When we tried to analyze them, we found these scenarios quite vague since on one hand, they do not express which is the title and intention of the scenario in a clear way, and, on the other hand, they do not provide a quantifiable response measure.

This was the reason why a first approach to build the utility tree was quite imprecise. Thus, we decided to meet the architecture designers to find out what they meant in their scenarios.

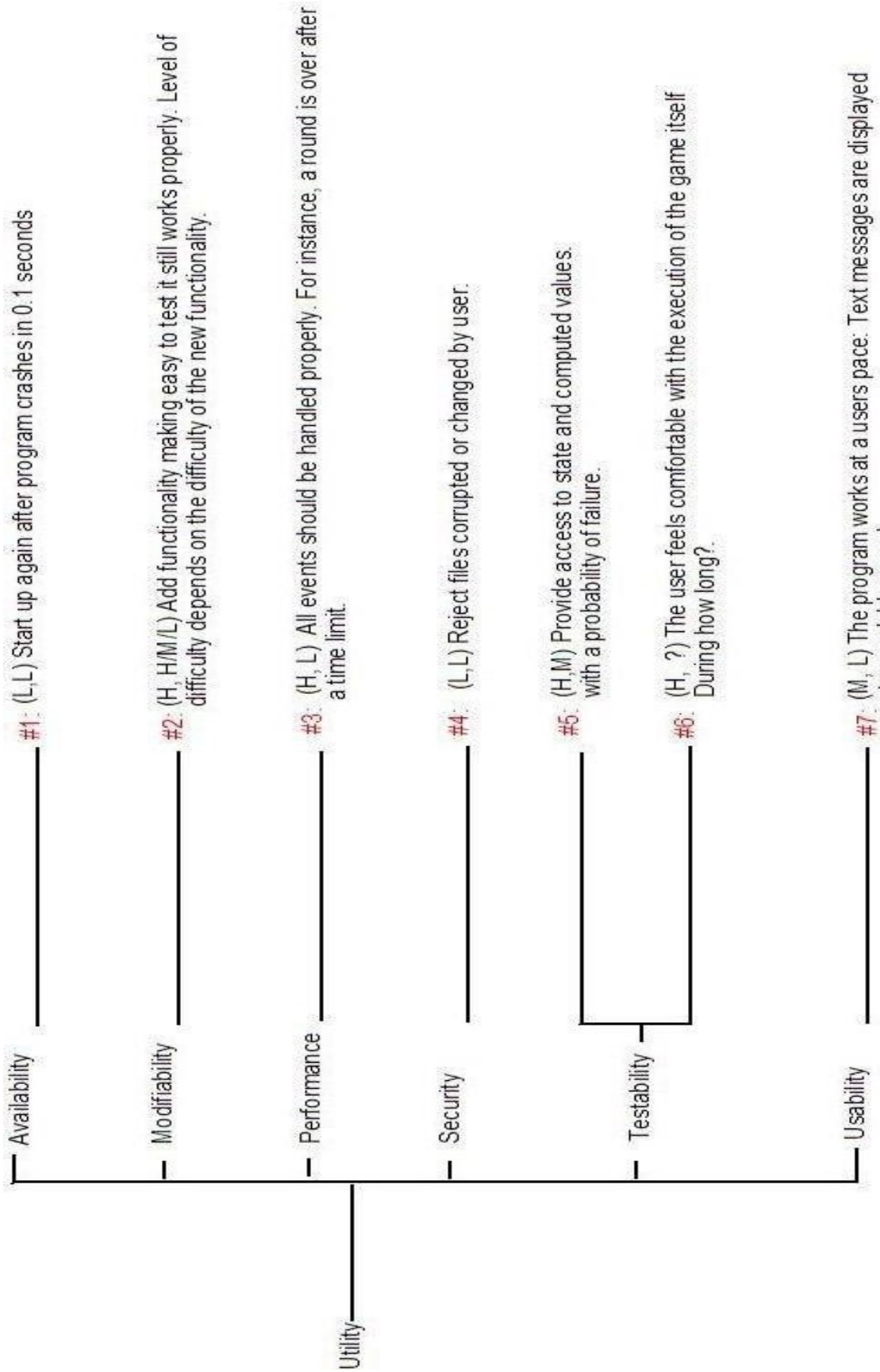
As result, the Utility tree showed in Illustration 1 has been built. There are some differences from the provided scenarios which became a little bit clearer after having met the architecture designers.

The scenarios are prioritized in the following way:
#6-#5-#2-#3-#7-#4-#1

S. #	Problems with Scenario meaning	Problems with Response Measure	Architecture Designers Explanation	Comments from evaluation team.
1	“Crash in system processors” was not clear for the evaluation team	Availability time is not quantifiable. No limits fixed	The game crashes for any reason. Time to start the game up once again less than 0.1 seconds	Availability is not a big issue for a video game => (L,L)
2	“Add functionality” does not say much. Not concrete.	No way to measure how this new functionality affects other functions is provided.	Designers more focused on make easier to test the other functions are still working when new functions are added than in specifying which functionality is added.	It is important, but difficulty depends on the difficulty of the new functionality => (H, H/M/L)
3	“Periodic events arrive” not concrete enough. What kind of events? User events? Game events?	“Latency” is not bounded in terms of time.	Designers state everything has to work properly , for instance, a round is over because of a time limit and another round must start straight afterwards.	Everything must work properly (following the logic of the program) is,of course, important, but is it really related to performance? => (H, L)
4	“User tries to change data” and access is granted. Not clear where the security was, since it seems access is always	How to measure data is damaged is not stated.	Designers explain that if reading from a modified file, something is wrong, that file is not loaded.	Security is not a big issue in a video game => (L, L)

	granted.			
5	Meaning is clear	Probability and time are not bounded. Failure of what?	Time to perform tests is dropped. Probability of failure in access was meant.	Failure in access seems to be unlikely => (H, M)
6	Meaning is clear	Measure is imprecise. What's the meaning of executable statements executed? Again no bounds provided in terms of time.	Designers explain they meant “playability” as the user is able to play with the game.	A measure in time is still missing. Play during 5 minutes is not the same as playing during 2 days. => (H, ?) Difficulty depends on time
7	Meaning is clear	What is “work at users pace”? How to measure user satisfaction?	An example about how text messages are shown at an appropriate speed in order to be read is provided.	It is relatively important and easy to achieve. => (M, L).

Illustration 1



Analysis of architectural approach

Analysis of Architectural Approach				
Scenario #: 3	Scenario: “Events should be handled properly”			
Attribute(s): Performance				
Environment: Normal mode				
Stimulus: Processes Stimuli				
Response: Latency				
Architectural Decisions	Sensitivity	Tradeoff	Risk	Nonrisk
Using XNA	S4	T2		
Using XQUEST		T2		

Analysis of Architectural Approach	
Scenario #: 6	Scenario: “Client acceptance tester”
Attribute(s): Testability	
Environment: Deployment time	
Stimulus: System delivered	

Response: Prepares test environment				
Architectural Decisions	Sensitivity	Tradeoff	Risk	Nonrisk
Using XNA				N1
Using XQUEST				N2

Analysis of Architectural Approach				
Scenario #: 5	Scenario: “Accessing state values”			
Attribute(s): Testability				
Environment: Development time				
Stimulus: Access to state values per Class				
Response: Provides access to state values and computed values				
Architectural Decisions	Sensitivity	Tradeoff	Risk	Nonrisk
Using XNA	S1	T1		
Using XQUEST	S2			N3

Analysis of Architectural Approach	
Scenario #: 5	Scenario: “Adding functionality”

Attribute(s): Modifiability				
Environment: Build Time				
Stimulus: Add functionality				
Response: Test modification and deploys modification				
Architectural Decisions	Sensitivity	Tradeoff	Risk	Nonrisk
Using XNA	S3	T2		
Using XQUEST	S3			

Analysis of Architectural Approach				
Scenario #: 7	Scenario: “End user wants to feel comfortable”			
Attribute(s): Usability				
Environment: runtime				
Stimulus: End user wants to feel comfortable				
Response: Display system state and work at users pace				
Architectural Decisions	Sensitivity	Tradeoff	Risk	Nonrisk
Using XNA	S5			
Using XQUEST				N3

Sensitivity points

S1: XNA may not allow access to internal functions, making difficult to access to internal values.

S2: XQUEST may not allow access to internal functions, making difficult to access to internal values.

S3: XQUEST and XNA was created to help developers to save time in game development.

S4: In this small scale, performance hit from using XNA should not be an issue and it will help to make it easier to guarantee performance.

S5: XNA provides ways to control timing and content displayed.

Tradeoffs

T1: It might be hard or impossible to have access to XNA internal state values but S3.
T2: Using XNA it is easier to add functionality (it allows you to focus on functionality), but it would be an upper limit for high end performance.

Risks and non-risks

N1: It does not influence the acceptance tests.

N2: It does not influence the acceptance tests.

N3: There is not any influence in usability using XQUEST.

Any risk was detected.

Own experiences of using ATAM

We consider that the projects are very small in order to apply a method so exhaustive as ATAM, so sometimes is difficult to apply the different steps with such an small quantity of information.
This strictness can be even bigger in the case of the video game, because the use of XNA imposes many of the architectural decisions.

Problems and issues

Although the ATAM-process by and all has been a very good learning experience for us all, we must say it has in no way been only positive. The subject matter is so new to all of us that it has been very difficult to understand it all quickly enough. The main problem was of course that we were in a way forced to start working on our documentation before we really had a chance to understand the subject matter fully.

It has definitely been a learning process, and by working through the problems we got a better handle on it in the end, but it was a long and hard road at times. I think since neither of our two groups had a very good grip on the whole process before we started we sort of had to interpret each others intentions a lot more than analyze each others documentations.

It should be mentioned that we had our share of missed appointments in our group and this also proved a serious challenge, but in the end we believe we managed fine. Even though it's never fun to get stuck with the ironing out of the report on the last night.

All in all we guess we would have wished there had been a bit more help with the original documentation before the ATAM-meeting so we had had a better understanding of what the document should properly contain and how it should be structured. This probably would have saved us some extra work from having to interpret the other groups understanding of how it should be done.

Change log

10th March 2008: First version of the documentation finished and delivered.