Trivial_i1a Documentation

Mario Germánico Pascual Tomás
Daniel Gonzalez Gómez
Edgar Herrero Uría
Adrián Prada Gil
Ainhoa Longo Pérez
Erik Sandsten

DEVELOPMENT OF THE TRIVIAL GAME

Proposal of the problem

The Game Company *NoGame* is going to start a new product line based on question/answer games that should work in several platforms. As a first step, they want to build a variant of the Trivial Pursuit game [1], although the goal is that other similar games based on question/answers could be created in the future.

In the first stage, they need to obtain a large number of questions to feed the different games. To that end, they want to build a simple application that reads files that contain the questions/answers, processes those files reporting any errors, and stores them in a database.

The files with the questions/answers will come from existing test banks. One of those test banks uses the GIFT format [2]. The application will use an internal representation of the questions using JSON format which will facilitate its storage in the database. The Company is planning to use MongoDb [3], although this decision has not yet been taken.

It is expected that the application can be executed in two phases. The first phase parses the files with the questions and generates the internal representation. The second phase takes the questions in that internal representation and stores them in the database. The execution of these 2 phases will be controlled by a company operator who could decide when he wants to execute each phase and even could automate that execution to be done at some time intervals.

It is important to warrant that the conversion is correct, easing the load process debugging. To that end, the solution that is being sought doesn't require an interactive interface but it requires several input options (select name of file to load, input format, name of output file, output format, etc.) and should enable to check that the internal representation of the conversion does not produce errors.

As it is planned to automate the conversion process to be executed at some specific times, the company doesn't require the application to be very time efficient or that the conversion process can be done interactively.

Although in this first step, the company only requires the conversion from GIFT format, the company is considering that the solution should allow other formats in the future. For example, the company is considering to process also questions in XML format like QTI [3].

Identification of the stakeholders

Trivial Development team:

They are the ones in charge of creating the trivial game and develop the architecture in order to facilitate the creation of future trivia games.

Some of their objectives are:

- Develop the Trivial game in an efficient way, controlling the costs and following the standards of usability thinking in the final user.
- Create a reusable architecture that can be implemented in other games based on question/answer mechanisms.

People in charge of NoGame:

These are the directors of the corporation, in charge of the budget, from which they allocate funds for the project and administrating the different games developed.

Some of their objectives are:

- The duration of the project should be short and the costs as minimum as possible.
- Get the maximum Profit.
- Get an application whose architecture can be used in other variants of the game.

Development team of NoGame:

They are the ones in charge of developing future trivia game that will use the architecture developed by Trivial Development team as a help in the process.

Some of their objectives are:

- Use that architecture to facilitate the development of new versions of the game.
- Finding the best technological alternatives to implement the game, and communicating them to the Trivial Development team.

Players of the Game:

The final users of the product, they want to play the game in an easy and intuitive way.

Some of their objectives are:

- Playing the game in an easy way without having to spend too much time trying to understand how the application works.

Identification of the quality attributes

Availability

- The system must have available all the functionality 24-7(every day at any time).

Modifiability

- Scalability of the system, we probably may do changes in the program.
- Facility to change the parser system, the questions and answers files format.
- Facility to change the connection with the database if in the future we want to change the database administrator program.

Performance

- We have to make sure that our systems responds to the user in a reasonable amount of time.
- Latency should be as low as possible in order to provide the user a good experience within the application.
- Throughput (number of events that take place within a given amount of time) should be as high as possible.

Security

- We have to ensure the security and strength of the system.

- We have to provide a good architecture in order to avoid non-ethical attack against our system that try to break into the server or that try to mislead the system.
- The system should be able to avoid DDoS attacks (very important in an application like this one).

Usability

- The system must be the most intuitive as possible, in order to help the user to understand its functioning.
- A manual must be available for any user that doesn't understand how to use the system.

Adaptability

- The system must work for any operating system.

Time to market

- Short development cycle

Cost-Benefit

- Null development cost

First approach to the solution

First approach

Once we have analyzed the requirements of the system, we have arrived to a possible solution based on a Batch system. That is a way of reducing the interation with the user avoiding an extra cost in specialized operators of the system focused on managing the translation and the databe. This solution will be formed of an application that will process text files with different formats extracting its content and storing it in a database.

Risks related to the solution

1. Guarantee of the database integrity in case of erroneous data introduce.
2. Protection against the introduction of not valid data both the parser and the database.

To avoid the problems caused by the risks mentioned, the following solutions are proposed:

1. Creating a backup of the database.
2. Creating a series of preconditions and exceptions that controls the flow of data that enters the system.

Quality attributes and stakeholders

Stakeholders vs Attributes	ST-01	ST-02	ST-03
AT001		X	
AT002	X		X
AT003	X		X
AT004			X
AT005		X	X
AT006		X	X
AT007		X	X
AT008	X	X	X
AT009	X	X	X
AT010	X	X	X
AT011		X	X
AT012			X
AT013			X
AT014	X	X	
AT015	X	X	

Business description of the solution

In our application we will have 2 differentiated systems

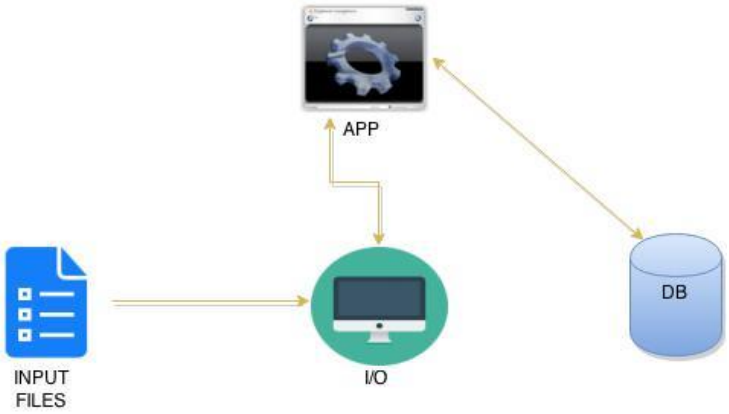
The Parser:

This section of the system will be the one in charge of reading and processing the input files with the questions and send it to the database.

It will have the responsibility of taking care that the format of file is respected and will rise and exception it case it contains non fitting information.

The Database:

Will store the information formatted by the parser until the future games will request it.

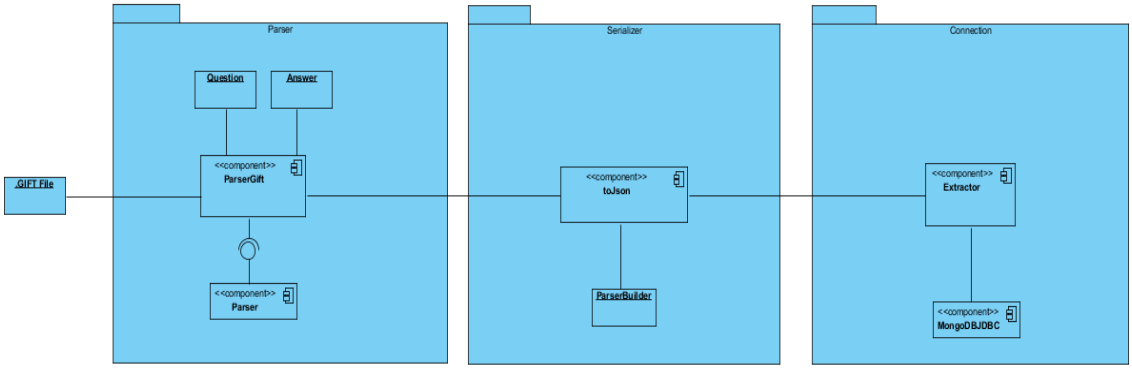


Quality scenarios

Scenario Number	Source of the stimulus	Stimulus	Environment	Artifact	Response	Response Measure	Affected quality attribute
1	Process the file questions	Data Processing	Explotation	Parser	Effective process time	Effective time > 90%	AT001
2	Modify the way the question is processed	Change the questions processing system.	Explotation	Parser	Sustitute the algorithm without altering other parts of the system	Other parts of the system altered = 0	AT002
3	Change the lexical analizer	Sustitute the lexical analyzer of the questions	Development	Parser	Sustitute the parser without altering other parts of the system	Parts of the system altered = 0	AT003
4	Change the target database	Introduce a new database system	Communication with the database.Development	Database system	Sustitute the database without altering other parts of the system	Parts appart from the database altered = 0	AT004
5	Acess to the Database processed data	Taking a question/answer from the database	Communication with the database	Database system	The answer/question is given from the database	Amount of data losed < 1%	AT005
6	Acess to the Database processed data	Perform the communication among the parser, the I/O and the database	Communication with the database	Database system	The resquested data is given to the other parts of the application	Request time< 5 s	AT006
7	Illigal access to the database	A user request data from de database without permissions	Communication with the database	Database system	Restricton is raised to avoid this access	Successfull illegal accesses < 1%	AT007
8	Acess to the raw documents	A user request data from de question/answer storage with no permissions	Illegal Explotation	Questions storage system	Restricton is raised to avoid this access	Successfull illegal accesses < 1%	AT008
9	Excess of access solicitude	A big amount of users request data from the data base.	Communication with the database	Database system	Find the origin of the massive access and lock it	Maximum number of allowed access attempts per user = 10	AT009
10	Introducing "help" in command line	Wrting the string "help" in the command line	Explotation	Command line	The help is shown in the command line	Availability = 100%	AT010
11	Changes in the system setup of the development team.	The operating system is changed.	Development	Development systems and teams	Sustitute the operating system without altering other parts of the system	Capabilities lost after the operating system changed = 0%	AT011
12	System in development	End of the development	Development	Development systems and teams	Low development time	Development time < 3 weeks	AT012
13	System in development	Cost of the development	Development	Development systems and teams	Low development cost	Total cost=0	AT013

Component Diagram

ASW Trivial



Summary

Name	Description
Parser	Parser package that reads the gift.
Serializer	Part in charge of transforming the information processed by the parser
Connection	Part in charge of the connection to the database.
Question	Question class in charge of getting the questions.
Answer	Answer class in charge of getting the answers.
ParserGift	This is the parser instance in charge of analyse gift files.
Extractor	Manage all the interactions with the data.
toJson	Converts every question and answer into JSON.
.GIFT File	It is a file that contains all the questions and answers.
Parser	Parser interface which goal is adapt the application to other formats.
ParserBuilder	Manage the upload of data to the database
MongoDBJDBC	Performs the connection with the database.

Details

Parser

Name	Value
Description	Parser package that reads the gift
Abstract	false
Leaf	false
Root	false
Visibility	public
Project Management	Name: Author — Value: Trivial_i1a Name: Create Date Time — Value: 25-feb-2015 20:30:32 Name: Last Modified — Value: 04-mar-2015 23:02:24

Children

Name	Description
Question	Question class in charge of getting the questions
Answer	Answer class in charge of getting the answers.
ParserGift	This is the parser instance in charge of analyse gift files.
N/A	Interface relation between the Parser and the ParserGift
Parser	Parser interface which goal is adapt the application to other formats

Serializer

Name	Value
Description	Part in charge of transforming the information processed by the parser
Abstract	false
Leaf	false
Root	false
Visibility	public

Project Management	Name	Value
	Author	Trivial_i1a
	Create Date Time	04-mar-2015 22:14:37
	Last Modified	04-mar-2015 23:12:24

Children

Name	Description
toJson	Converts a every question and answer into JSON
ParserBuilder	Manage the data upload to the database

Connection

Name	Value
Description	Part in charge of the connection to the database.
Abstract	false
Leaf	false
Root	false
Visibility	public
Project Management	Name: Author / Value: Trivial_i1a Name: Create Date Time / Value: 25-feb-2015 20:36:12 Name: Last Modified / Value: 04-mar-2015 23:12:24

Children

Name	Description
Extractor	Manage all the interactions with the data.
MongoDBJDBC	Performs the connection with the database.

Question

Name	Value
Description	Question class in charge of getting the

	questions
Visibility	Unspecified
Project Management	**Name** / **Value** Author / Trivial_i1a Create Date Time / 04-mar-2015 22:24:15 Last Modified / 04-mar-2015 23:02:24

Relationships

Unnamed Generic Connector	
From	ParserGift
Project Management	**Name** / **Value** Author / Trivial_i1a Create Date Time / 04-mar-2015 22:25:41 Last Modified / 04-mar-2015 22:29:48

Answer

Name	Value
Description	Answer class in charge of getting the answers.
Visibility	Unspecified
Project Management	**Name** / **Value** Author / Trivial_i1a Create Date Time / 04-mar-2015 22:24:21 Last Modified / 04-mar-2015 23:02:24

Relationships

Unnamed Generic Connector	
From	ParserGift
Project Management	**Name** / **Value** Author / Trivial_i1a Create Date Time / 04-mar-2015 22:25:35 Last Modified / 04-mar-2015 22:29:48

ParserGift

Name	Value
Description	`This is the parser instance in charge of analyze gift files.`
Active	false
Business Key Mutable	true
Business Model	false
Visibility	public
Abstract	false
Leaf	false
Root	false
Indirectly Instantiated	true
Project Management	**Name** / **Value** Author: Trivial_i1a Create Date Time: 25-feb-2015 20:30:51 Last Modified: 04-mar-2015 23:02:24

Relationships

Unnamed Association	
To	**Name** / **Value** End Model Element: toJson Provide Property Getter Method: false Provide Property Setter Method: false Multiplicity: Unspecified Visibility: Unspecified Aggregation Kind: None Navigable: Navigable Derived: false Derived Union: false Read Only: false Static: false Leaf: false Type: toJson

	Project Management	**Name**	**Value**
		Author	Trivial_i1a
		Create Date Time	25-feb-2015 20:46:00
Abstract	false		
Leaf	false		
Visibility	Unspecified		
Derived	false		
Project Management	**Name**	**Value**	
	Author	Trivial_i1a	
	Create Date Time	25-feb-2015 20:46:00	
	Last Modified	04-mar-2015 22:20:50	

Unnamed Generic Connector		
To	Answer	
Project Management	**Name**	**Value**
	Author	Trivial_i1a
	Create Date Time	04-mar-2015 22:25:35
	Last Modified	04-mar-2015 22:29:48

Unnamed Generic Connector		
To	Question	
Project Management	**Name**	**Value**
	Author	Trivial_i1a
	Create Date Time	04-mar-2015 22:25:41
	Last Modified	04-mar-2015 22:29:48

Unnamed Usage		
To	N/A	
Visibility	Unspecified	
Project Management	**Name**	**Value**
	Author	Trivial_i1a
	Create Date Time	04-mar-2015 22:40:06
	Last Modified	04-mar-2015 22:42:23

Unnamed Generic Connector

From	.GIFT File	
Project Management	**Name**	**Value**
	Author	Trivial_i1a
	Create Date Time	04-mar-2015 22:29:27
	Last Modified	04-mar-2015 22:29:48

Extractor

Name	Value	
Description	`Manage all the interactions with the data.`	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Abstract	false	
Leaf	false	
Root	false	
Indirectly Instantiated	true	
Project Management	**Name**	**Value**
	Author	Trivial_i1a
	Create Date Time	25-feb-2015 20:41:38
	Last Modified	04-mar-2015 23:22:25

Relationships

Unnamed Association		
To	**Name**	**Value**
	End Model Element	MongoDBJDBC
	Provide Property Getter Method	false
	Provide Property Setter Method	false
	Multiplicity	Unspecified
	Visibility	Unspecified
	Aggregation Kind	None
	Navigable	Navigable
	Derived	false

	Derived Union	false
	Read Only	false
	Static	false
	Leaf	false
	Type	MongoDBJDBC
	Project Management	**Name** / **Value** Author: Trivial_i1a Create Date Time: 25-feb-2015 20:46:10
Abstract	false	
Leaf	false	
Visibility	Unspecified	
Derived	false	
Project Management	**Name** / **Value** Author: Trivial_i1a Create Date Time: 25-feb-2015 20:46:10 Last Modified: 25-feb-2015 20:48:09	

Unnamed Association

From	End Model Element	toJson
	Provide Property Getter Method	false
	Provide Property Setter Method	false
	Multiplicity	Unspecified
	Visibility	Unspecified
	Aggregation Kind	None
	Navigable	Navigable
	Derived	false
	Derived Union	false
	Read Only	false
	Static	false
	Leaf	false
	Type	toJson
	Project Management	**Name** / **Value** Author: Trivial_i1a

		Create Date Time	25-feb-2015 20:46:06
Abstract	false		
Leaf	false		
Visibility	Unspecified		
Derived	false		
Project Management	**Name**	**Value**	
	Author	Trivial_i1a	
	Create Date Time	25-feb-2015 20:46:06	
	Last Modified	04-mar-2015 22:20:50	

toJson

Name	Value	
Description	`Converts a every question and answer into JSON`	
Active	false	
Business Key Mutable	true	
Business Model	false	
Visibility	public	
Abstract	false	
Leaf	false	
Root	false	
Indirectly Instantiated	true	
Project Management	**Name**	**Value**
	Author	Trivial_i1a
	Create Date Time	25-feb-2015 20:31:33
	Last Modified	04-mar-2015 23:12:24

Relationships

Unnamed Association		
To	**Name**	**Value**
	End Model Element	Extractor
	Provide Property Getter Method	false
	Provide Property Setter	false

	Method	
	Multiplicity	Unspecified
	Visibility	Unspecified
	Aggregation Kind	None
	Navigable	Navigable
	Derived	false
	Derived Union	false
	Read Only	false
	Static	false
	Leaf	false
	Type	Extractor
	Project Management	**Name** / **Value** Author / Mario Create Date Time / 25-feb-2015 20:46:06
Abstract	false	
Leaf	false	
Visibility	Unspecified	
Derived	false	
Project Management	**Name** / **Value** Author / Mario Create Date Time / 25-feb-2015 20:46:06 Last Modified / 04-mar-2015 22:20:50	

Unnamed Association

From	**Name**	**Value**
	End Model Element	ParserGift
	Provide Property Getter Method	false
	Provide Property Setter Method	false
	Multiplicity	Unspecified
	Visibility	Unspecified
	Aggregation Kind	None
	Navigable	Navigable
	Derived	false
	Derived Union	false
	Read Only	false

	Static	false
	Leaf	false
	Type	ParserGift
	Project Management	**Name** / **Value** Author / Trivial_i1a Create Date Time / 25-feb-2015 20:46:00
Abstract	false	
Leaf	false	
Visibility	Unspecified	
Derived	false	
Project Management	**Name** / **Value** Author / Trivial_i1a Create Date Time / 25-feb-2015 20:46:00 Last Modified / 04-mar-2015 22:20:50	

Unnamed Generic Connector	
From	ParserBuilder
Project Management	**Name** / **Value** Author / Trivial_i1a Create Date Time / 04-mar-2015 22:32:47 Last Modified / 04-mar-2015 22:42:23

.GIFT File

Name	**Value**
Description	`It is a file that contains all the questions and answers.`
Visibility	Unspecified
Project Management	**Name** / **Value** Author / Trivial_i1a Create Date Time / 25-feb-2015 20:32:02 Last Modified / 04-mar-2015 22:52:23

Relationships

Unnamed Generic Connector	
To	ParserGift
Project Management	

Name	Value
Author	Trivial_i1a
Create Date Time	04-mar-2015 22:29:27
Last Modified	04-mar-2015 22:29:48

N/A

Name	Value
Active	false
Business Key Mutable	true
Business Model	false
Visibility	public
Leaf	false
Root	false
Stereotypes	Interface
Project Management	

Name	Value
Author	daniel
Create Date Time	04-mar-2015 22:40:00
Last Modified	04-mar-2015 22:42:23

Relationships

Unnamed Realization	
To	Parser
Visibility	Unspecified
Project Management	

Name	Value
Author	Trivial_i1a
Create Date Time	04-mar-2015 22:40:00
Last Modified	04-mar-2015 22:42:23

Unnamed Usage	
From	ParserGift
Visibility	Unspecified

Project Management	Name	Value
	Author	Trivial_i1a
	Create Date Time	04-mar-2015 22:40:06
	Last Modified	04-mar-2015 22:42:23

Parser

Name	Value
Description	`Parser interface which goal is adapt the application to other fomats`
Active	false
Business Key Mutable	true
Business Model	false
Visibility	public
Abstract	false
Leaf	false
Root	false
Indirectly Instantiated	true

Project Management

Name	Value
Author	Trivial_i1a
Create Date Time	04-mar-2015 22:26:25
Last Modified	04-mar-2015 23:12:24

Relationships

Unnamed Realization	
From	N/A
Visibility	Unspecified

Project Management

Name	Value
Author	Trivial_i1a
Create Date Time	04-mar-2015 22:40:00
Last Modified	04-mar-2015 22:42:23

ParserBuilder

Name	Value
Description	Manage the data upload to the database
Visibility	Unspecified
Project Management	**Name** / **Value** Author: Trivial_i1a Create Date Time: 04-mar-2015 22:32:31 Last Modified: 04-mar-2015 23:12:24

Relationships

Unnamed Generic Connector	
To	toJson
Project Management	**Name** / **Value** Author: Trivial_i1a Create Date Time: 04-mar-2015 22:32:47 Last Modified: 04-mar-2015 22:42:23

MongoDBJDBC

Name	Value
Description	Performs the connection with the database.
Active	false
Business Key Mutable	true
Business Model	false
Visibility	public
Abstract	false
Leaf	false
Root	false
Indirectly Instantiated	true
Project Management	**Name** / **Value** Author: Trivial_i1a Create Date Time: 25-feb-2015 20:42:05 Last Modified: 04-mar-2015 23:22:25

Relationships

<table>
<tr><th colspan="2">Unnamed Association</th></tr>
<tr><td>From</td><td>
<table>
<tr><th>Name</th><th>Value</th></tr>
<tr><td>End Model Element</td><td>  Extractor</td></tr>
<tr><td>Provide Property Getter Method</td><td>false</td></tr>
<tr><td>Provide Property Setter Method</td><td>false</td></tr>
<tr><td>Multiplicity</td><td>Unspecified</td></tr>
<tr><td>Visibility</td><td>Unspecified</td></tr>
<tr><td>Aggregation Kind</td><td>None</td></tr>
<tr><td>Navigable</td><td>Navigable</td></tr>
<tr><td>Derived</td><td>false</td></tr>
<tr><td>Derived Union</td><td>false</td></tr>
<tr><td>Read Only</td><td>false</td></tr>
<tr><td>Static</td><td>false</td></tr>
<tr><td>Leaf</td><td>false</td></tr>
<tr><td>Type</td><td>  Extractor</td></tr>
<tr><td>Project Management</td><td>
<table>
<tr><th>Name</th><th>Value</th></tr>
<tr><td>Author</td><td>Trivial_i1a</td></tr>
<tr><td>Create Date Time</td><td>25-feb-2015 20:46:10</td></tr>
</table>
</td></tr>
</table>
</td></tr>
<tr><td>Abstract</td><td>false</td></tr>
<tr><td>Leaf</td><td>false</td></tr>
<tr><td>Visibility</td><td>Unspecified</td></tr>
<tr><td>Derived</td><td>false</td></tr>
<tr><td>Project Management</td><td>
<table>
<tr><th>Name</th><th>Value</th></tr>
<tr><td>Author</td><td>Trivial_i1a</td></tr>
<tr><td>Create Date Time</td><td>25-feb-2015 20:46:10</td></tr>
<tr><td>Last Modified</td><td>25-feb-2015 20:48:09</td></tr>
</table>
</td></tr>
</table>

User´s system manual

General functionality

The command line usage of the application is:

```
java -jar extract.jar <mode> <parameters>
```

modes:

`-e <input-file> <input-format> <output-file>:` Extraction mode. The application is to process a file containing questions.

Parameters:

<input-file> - The filename containing the questions to be processed.

<input-format> - The format of the input file, which can only be *gift* for now.

<output-file> - The filename of the outputted file created by the application.

<output-format> - The format of the output file, which can only be *json* for now.

`-db <input-file> <server>`: Database storing mode.

Parameters:

<input-file> - The filename of the file with the JSON-formatted questions.

<server> - The server address, which is spelled ***address:port***. if left out is assumed to be localhost:27017.

GIFT format

// text	Comment until end of line (optional)
::title::	Question title (optional)
text	Question text (becomes title if no title specified)
{ ... =right ... }	Correct answer for multiple choice, (multiple answer? -- see page comments) or fill-in-the-blank
{ ... ~wrong ...	Incorrect answer for multiple choice or multiple answer
}	