Questionnaires and Surveys

Open-source web-app for questionnaires and surveys

---

*G*roup 1000_2:

Kevin Brotcke
Maria Carrasco
Andrew Furusawa
Joshua Papa

*V*ersion: 3.0

Winter 2011
University of California Irvine

Contents

1 Introduction

This document is the software requirements specification for an open source web application for questionnaires and surveys.

1.1 Purpose

This document outlines the online survey system describing its system requirements for the production of a functional system, as well as user requirements in order to produce software compliant with the user's needs. We define the assumptions used to create the system, and provide use-case diagrams representing the system functionality and relationships.

The following document is intended to be a third version of the requirements that must be formalized later.

1.1.1 Intended Audience and Reading Suggestions

This document is intended for developers, project managers, users, testers and documentation writers. The main sections contained are an *Introduction* that contains information and instructions about the document. An *Overall Description* that gives a general idea of the product being specified in this SRS. The *Requirements*, describing the characteristics of the software. And the *Use Case Diagram*, to pressent a graphical overview of the functionality provided by the system.

1.2 Definitions, acronyms, and abbreviations

SRS *Software Requirements Specification.*

DBMS *Data Base Management System*. The place where we will store the application data.

CSS *Cascading Style Sheets*. Style sheet language used to describe the presentation semantics (the look and formatting) of a document.

HTML *HyperText Markup Language*. Markup language for web pages.

PHP *Hypertext Preprocessor*. Scripting language.

DSL *Digital Subscriber Line*. Technology that provides digital data transmission over the wires of a local telephone network.

FTP *File Transfer Protocol*. Standard network protocol used to copy a file from one host to another over a network.

XLSX *File extension*. Excel Microsoft Office XML Format Spreadsheet.

URL *Uniform Resource Locator* Specifies where an identified resource is available.

1.3 References

Project Webpage *sites.google.com/site/inf117sprouse/home/*

IEEE Std 830-1998 *standards.ieee.org/findstds/standard/830-1998*

Repository *http://code.google.com/p/questionnaires-and-surveys/*

1.4 Overview

Jon Sprouse of the Department of Cognitive Sciences at UCI is studying grammatical theory through surveys–both online and in the field. These surveys require careful construction through colors, ordering, and other various textual attributes to provide useful data for his work. In order to assist his research, a web-based system that generates these surveys provides efficiency and automation that would otherwise be done manually and intensively.

2

Overall description

2.1 Product perspective

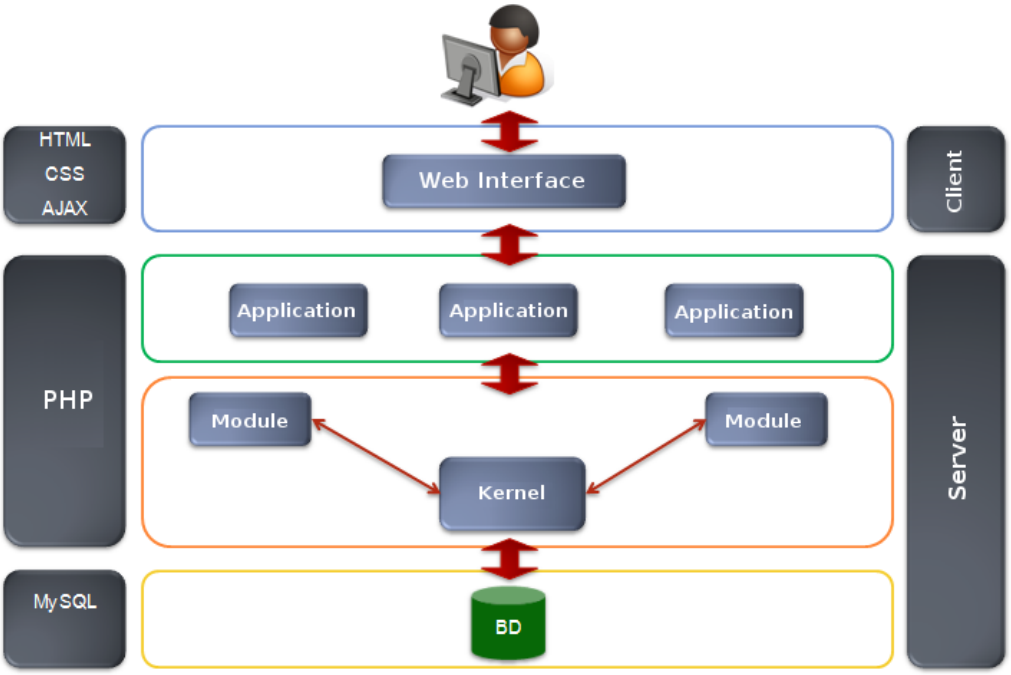
This product aims to fill an existing need. Although there are many different choices for online survey generators, such as limeSurvey or SurveyMonkey, none of these options seem to meet the requirements that are needed.

Developing into a web-based interface, the survey system is intended to contain customizable functionalities during survey creation, multiple version creation, and allow for survey participation. Customizations include both font attribute and layout manipulation, allowing the administrator to invoke different psychological responses from the participant. Multiple versions allow for the juxtaposition and analysis of participant responses with the use of "same-content, different-layout.[en]vironments.

2.1.1 System interfaces

The software will have an *admin interface*. This interface will allow the administrator to perform several tasks.

- Upload experiments.
- Download results.
- Change appearance.
    - Sample display at bottom that shows real time changes.



Caption : Block Diagram showing major components.

2.1.2 User interfaces

The main features that the software will present are:

- *Sets of surveys*, allowing the user to automatically loop through them.
- *Pseudo-Randomization* of the order of the surveys.
- *Result file*. This file will contain these main fields: Subject ID, survey number, code and responses.

2.1.3 Hardware interfaces

The software should run on a wide range of hardware. Please note that much of the computer equipment might have heterogeneous and unpredictable characteristics. Should the software run on a wide range of hardware. The web characteristics such as AJAX should not be abused because they do not work in older browsers and require more resources on the client side.

2.1.4 Software interfaces

The software should run on Windows. The software must be compatible with MS Excel and the main Internet web browsers such as Mozilla Firefox, Google Chrome and Internet Explorer.

2.1.5 Communications interfaces

The software must communicate with the clients' browsers via HTTP over TCP / IP.

2.1.6 Memory

There is no objective measure, but given the characteristics of the hardware described, it should be noted that the web portal should be light. There will not be more than two hundred surveys at the same time.

2.1.7 Operations

There are different roles in the organization that defines the operational mechanisms:

- The *Administrator* is responsible for uploading the surveys and questionnaires.
- The *Participant* can register in the web page. It should be able to configure its preferences and consult its surveys and results.

2.1.8 Site adaptation requirements

There are no site adaptation requirements, besides those already discussed above.

2.2 Product functions

Broadly, the main functions of the software will be:

- Surveys management by the administrator.
- Ability to answer the survey by the participants.

2.3 User characteristics

There are two different user types:

- *Administator*. Middle class people with different computer literacy. Assumptions:
  - Has the ability to transfer files over FTP.
  - Must have an Admin account to login to manage surveys.
  - The Administrator should be able to use the Control Panel to DL/UL surveys to/from the server.
  - Ability to edit xlsx files and understand how to apply attributes to the survey prior to upload.
- *Participant*. Young people who are very familiar with the technologies. They will have to access to the URL for a given experiment.

2.4 Assumptions and dependencies

These requirements have been developed by making some assumptions based on characteristics of users and the general descriptions of equipment on which it will use the software.

3 Requirements

3.1 External interfaces

3.1.1 User interface

- Should be a web interface.
- Should follow the W3C standards.
- Should follow AA accessibility standards.

3.1.2 Software interface

- Server installed with PHP 5.
- Internet connection with DSL or Cable bandwith.
- *xlsx* Microsoft Excel Version $\geq$ 07
- Latest version of Internet Explorer, Firefox, or Chrome as the Web browser.

3.2 Functional Requirements

3.2.1 Survey

- A set of surveys consist of an experiment.

- For a given experiment, all surveys must be evenly distributed randomly to the user
  - The user picks the experiment but is automatically assigned a survey
  - The user does not know which survey they are taking
- Must support pseudo randomization
  - The first letter of the survey code must not be next to another question with the same letter.
- Auto-complete must be disabled

Appearance Of Surveys

- the appearance of the sentences on the web form (font, size, color, spacing, location, etc) needs to be customizable through CSS/HTML tags
- The web form needs to accept a couple of different response types (radio buttons, text boxes, etc)
- Left/Center/Right justifiable
- Autocomplete off
- *CSS* template file option
- Sentences per page

Administrator

The administrator shall be able to do the following actions:

- Create Survey
  transfer a new xlsx file via ftp to an input folder
- Edit Survey
  overrides the existing xlsx file
- View Survey
  downloads the xlsx file from the server
- Download Results

Participant

The participant shall be able to perform the following actions:

- Request Survey by URL
  Correct survey automatically assigned to user based off URL.
- Submit Survey Response

3.2.2 Input

- Uses *xlsx* format as input file.
- Input is read from a file containing:
  - Code - A unique identifier used for analysis. Only the Administrator sees this.
  - Question - The actual question the user must answer.
  - Format - Can be either Radio Button or Text Box. A survey may consist of all of one type or a combination of the two.
- Options in input file:
  - Justification
  - Pagination
  - CSS file
  - Header Title
  - Table HTML Properties
  - Number of spaces between types of questions

3.2.3 Output

- The system shall have an option to export survey results.
- The fields that shall be present in the file are:
  - subject ID
  - survey number
  - code
  - responses

3.3 Performance requirements

- Surveys will be generated within 2 seconds.

3.4 Logical database requirements

The information stored in the database will be regarding to the experiments and surveys.

3.5 Software system attributes

3.5.1 Ease of use

- Installation and maintenance for the Administrator must be very easy.

3.5.2 Availability

The software should have no interruptions in service except the following conditions:

- updating some of the service dependencies: DBMS, HTTP server, Web Server or operating system.
- system re-installation.
- Update from older versions of software.

3.5.3 Portability

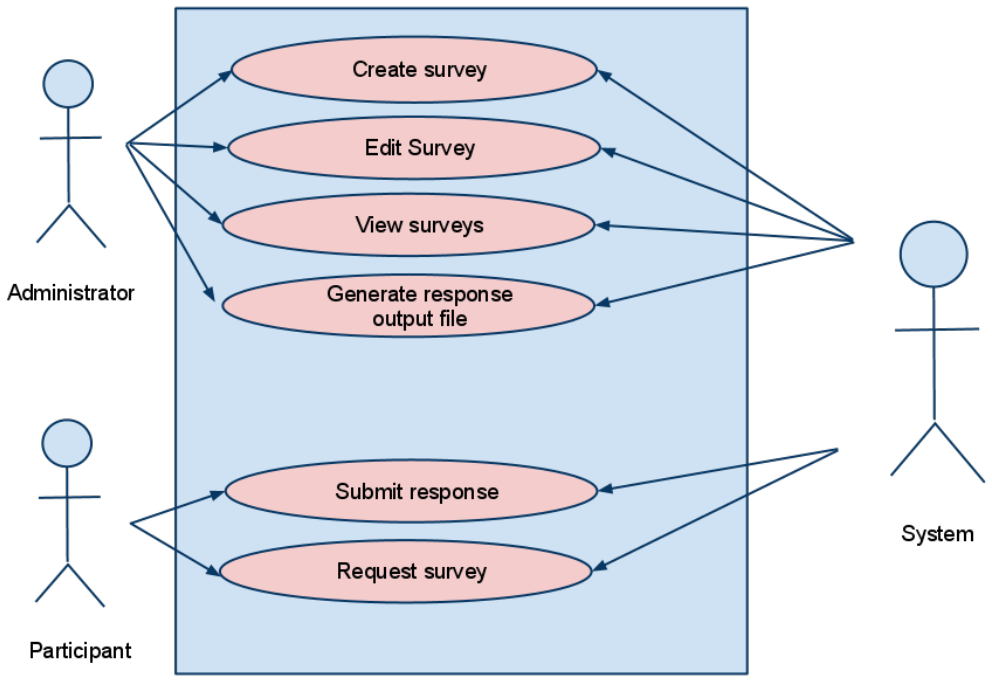
- The pages generated must be 100 % compatible
- To ensure compatibility with other browsers, XHTML 1.0 standard should be used for generated pages.

4

Use Cases

4.1 Introduction

Specified below are the different use cases of the application, that allows to study the interaction between user and system to achieve the functionality specified above.



Caption : Use Case Diagram

4.2 Use case list

4.3.1 Create Survey.

4.3.2 Manage Surveys

4.3.2 Edit Survey.

4.3.2 View Surveys.

4.3.3 Generate response output file.

4.3.5 Submit response.

4.3.4 Request Survey.

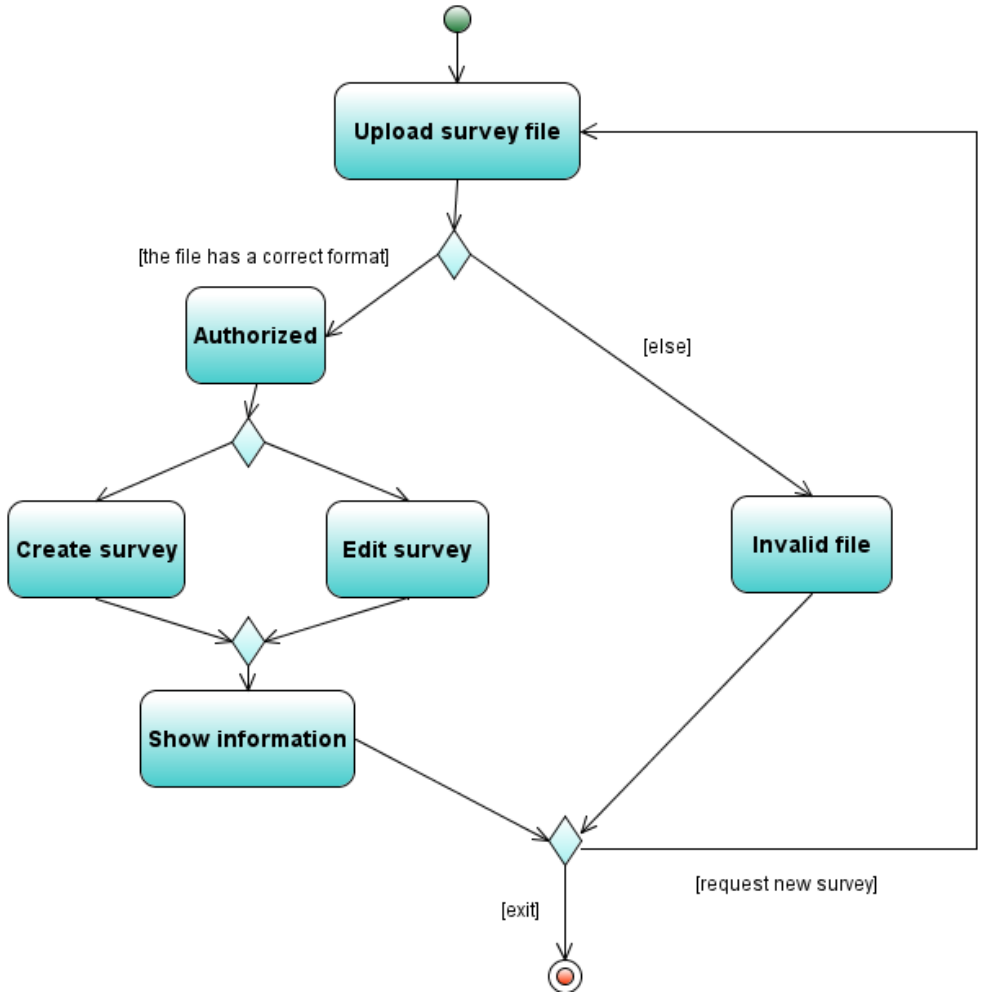
4.3 Use Cases

4.3.1 Create survey

Name	Create Survey
Description	The administrator of the application creates an experiment.
Dependencies	None.
Actors	Administrator
Preconditions	User should have access to the system.
Postconditions	A new survey will be created.

Basic Path

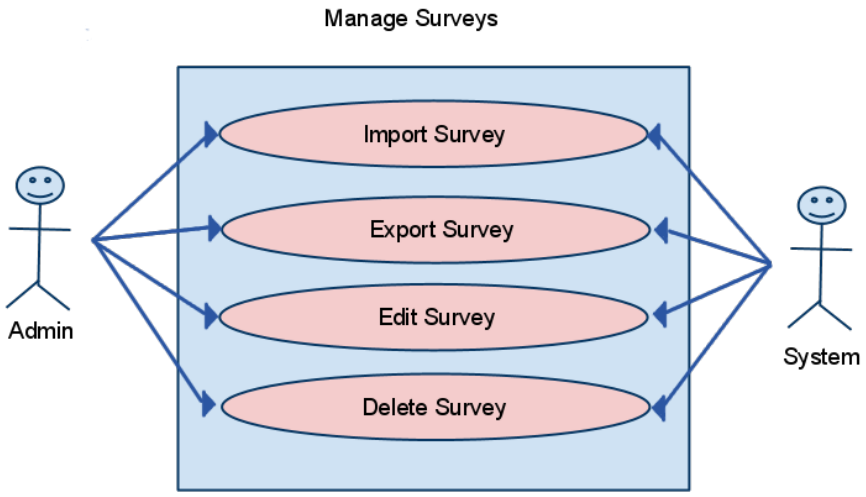
User	System
1.- Access to the option Create Survey, through the main menu.	2.- Shows an option to upload survey files.
3.- Select the file to upload and press OK.	4.- Checks the validity of data.
	5.- Confirms the creation of the survey.



Caption : Activity Diagram

Alternative Paths

User	System
	5b.- The data is incorrect. Asks for another file input.



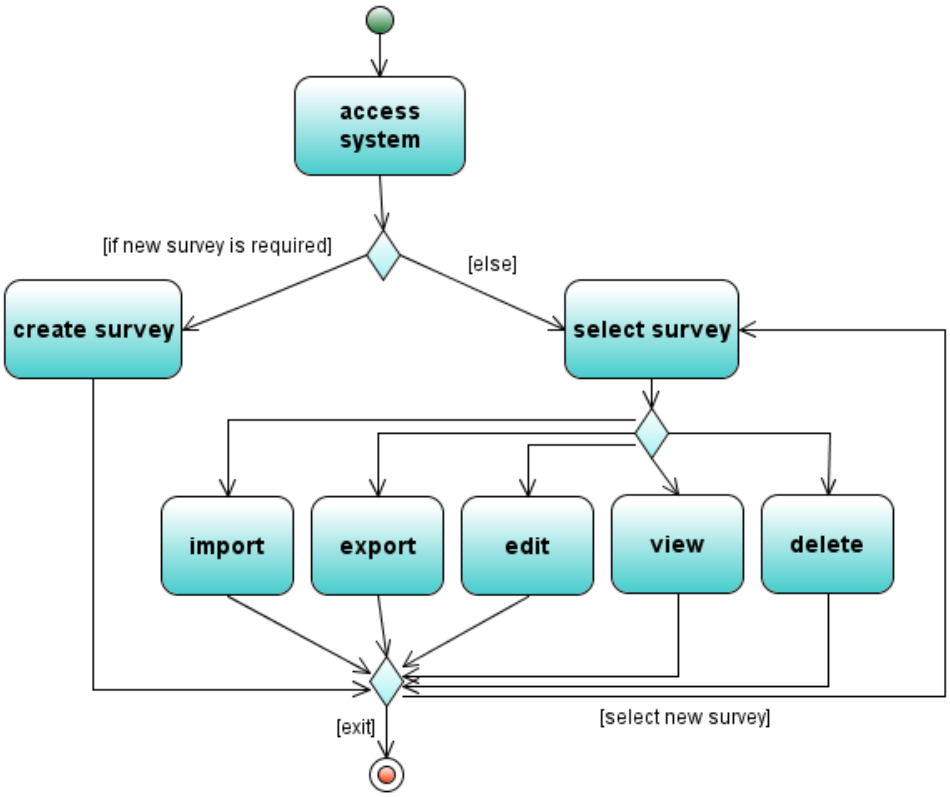
Caption : Use Case Diagram

4.3.2 Manage Surveys

Edit survey

Name	Edit Survey
Description	The user edits an existing survey.
Dependencies	4.3.1
Actors	Administrator
Preconditions	A survey must have been already created.
Postconditions	An existing survey will be changed.

Basic Path	
User	System
1.- Choose the option *Edit survey* in the menu.	2.- Shows the options for editing the file.
3.- Chage the options and press OK.	4.- Checks validity of data.
	5.- Confirms the edition of the survey.



Caption : Activity Diagram

Alternative Paths

User	System
	5b.- The data is incorrect. Returns to step 2.

View Surveys

Name	View Surveys
Description	The user views the existing surveys.
Dependencies	4.3.1
Actors	Administrator
Preconditions	The user must have access to the system.
Postconditions	The survey(s) will be displayed on the screen.

Basic Path

User	System
1.- Choose the option *View surveys*.	2.- Displays a menu, with all the current surveys in the system.
3.- Choose a survey within the given list.	4.- Shows the selected survey.

Alternative Paths

User	System
	2b.- There are no surveys in the system. Displays the message: *There are no surveys available.*
	4b.- If the systems fails while accessing the required data, it returns to step 2.

4.3.3 Generate Response Output File

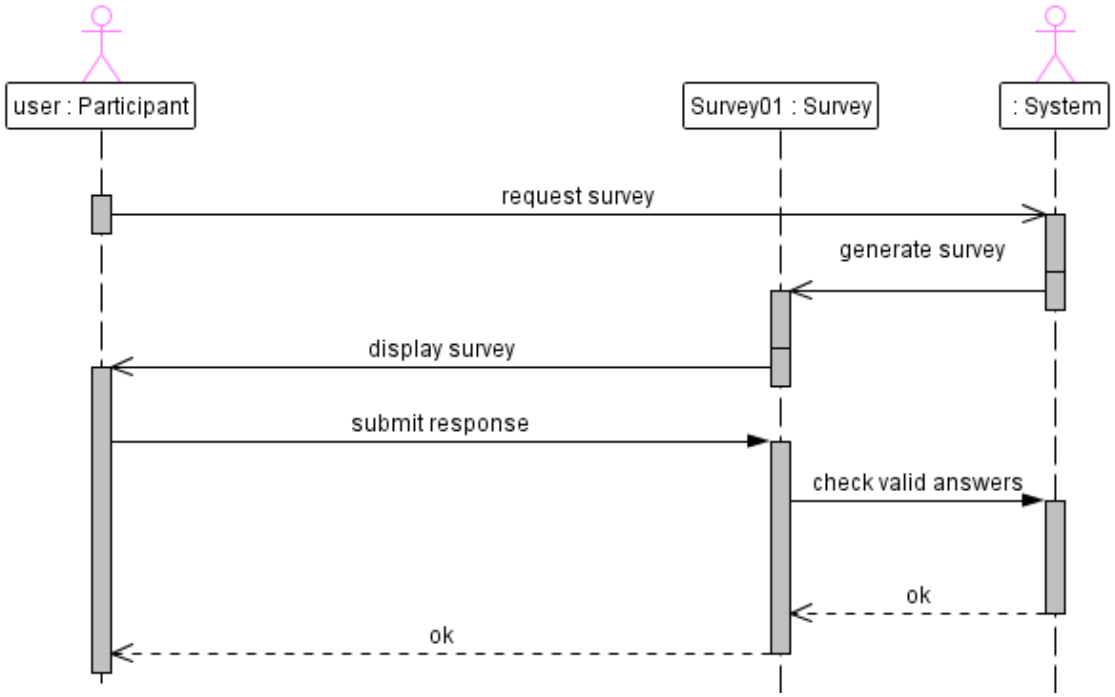
Name	Generate response output file.
Description	Generates an output file containing all surveys responses.
Dependencies	4.3.1 4.3.5
Actors	Administrator
Preconditions	A survey must have been already created and submited.
Postconditions	An output file will be created.

Basic Path

User	System
1.- Choose the option *Generate Response File*.	2.- Shows a list with all the current surveys.
3.- Selects a survey from which to obtain the output file and selects Ok.	4.- Displays a window with Saving options.
5.- Selects a path where to store the output file, and a name for the file. Clicks *Save*	6.- Performs the operation and displays an operation completed message.

Alternative Paths

User	System
	6b.- The operation could not be done. No file was generated. Returns to step 2.



Caption : Sequence Diagram

4.3.4 Request Survey

Name	Request Survey
Description	The user request a new survey to be displayed.
Dependencies	4.3.1
Actors	Participant
Preconditions	A survey must have been already created.
Postconditions	A survey will be displayed.

Basic Path

User	System
1.- Enters the URL in the web browser.	2.- Shows the main menu to the user.
3.- Selects the option *Request Survey*.	4.- Presents a list of surveys .
4.- Chooses a survey.	6.- Displays a version of the selected survey.

Alternative Paths

User	System

4.3.5 Submit Response

Name	Submit Response
Description	The user submits a fully completed survey.
Dependencies	4.3.4
Actors	Participant
Preconditions	A survey must have been already requested.
Postconditions	A survey response will be submitted.

Basic Path

User	System
1.- Choose the option *Submit Survey*.	2.- Stores the data and Shows a congratulations message.

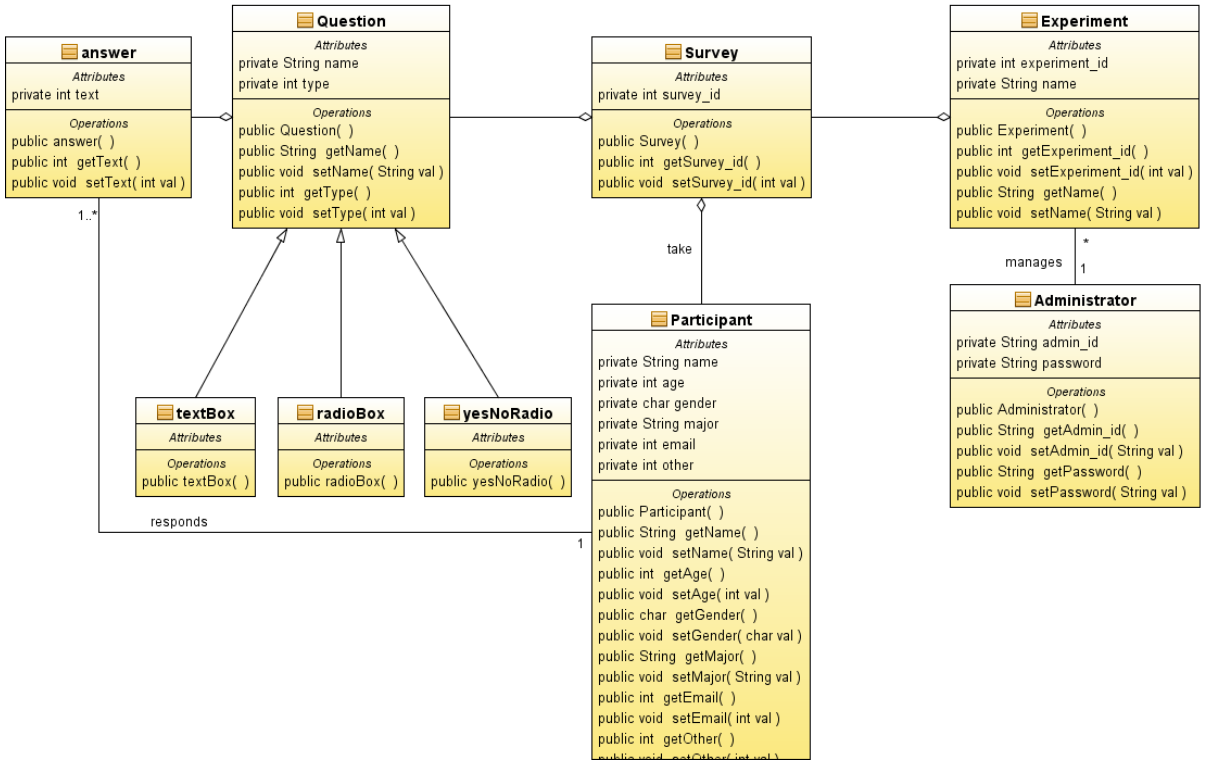
Alternative Paths

User	System
	2b.- If there are any incomplete box, shows the survey again.

5

Analysis

5.1 Class Diagram



Caption : Class Diagram

Bibliography

[1] Stevens, P. & Pooley, R., 1999. *USING UML. Software Engineering with objects and components.* 2^{nd}ed. Harlow Essex: Addison Wesley

[2] Britton, C. & Doake, J., 2005. *A student guide to OBJECT-ORIENTED DEVELOPMENT* .Burlington, MA: Elsevier Butterworth-Heinemann