Design Iteration 4

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

1.1 Purpose

This document describes the design layout for the Survey Generator program by Team 1000 for their Informatics 117 class during Winter 2011. Major design components such as the data design, architectural and component-level design, and user interface design will be discussed. This is meant as a reference for team members during implementation as well evidence for stakeholders.

1.2 Project Description

The project uses a database back-end to generate surveys on-the-fly using a web app. The main components consists of a SQLite database, HTML front-end of surveys and administrator panel, and business logic to generate the surveys.

2

Data Design

2.1 Database Description

The data will be stored in the form of an embedded SQLite database. (See figure 4.2)

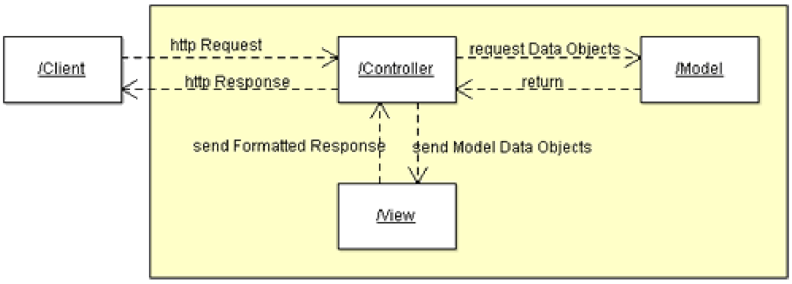
2.2 Temporary Data Structures

This will need to import and export XML files in the form XLSX as well as CSV files. These will only be temporary since they are being converted into and out of the SQLite database.

3

Design Pattern : MVC

For our application we decided to use the *Model View Controller* pattern. This decision was made based on the fact that this pattern is the most used for today's world web applications.



Caption : MVC Collaboration Diagram.

The *MVC* pattern divides our application in three main modules:

Model

This module is responsible for managing the data. It is responsible for storing and retrieving information from the database, in our case *SQLite*.

View

The view is the module that display the data provided by the model in a specific format.

Controller

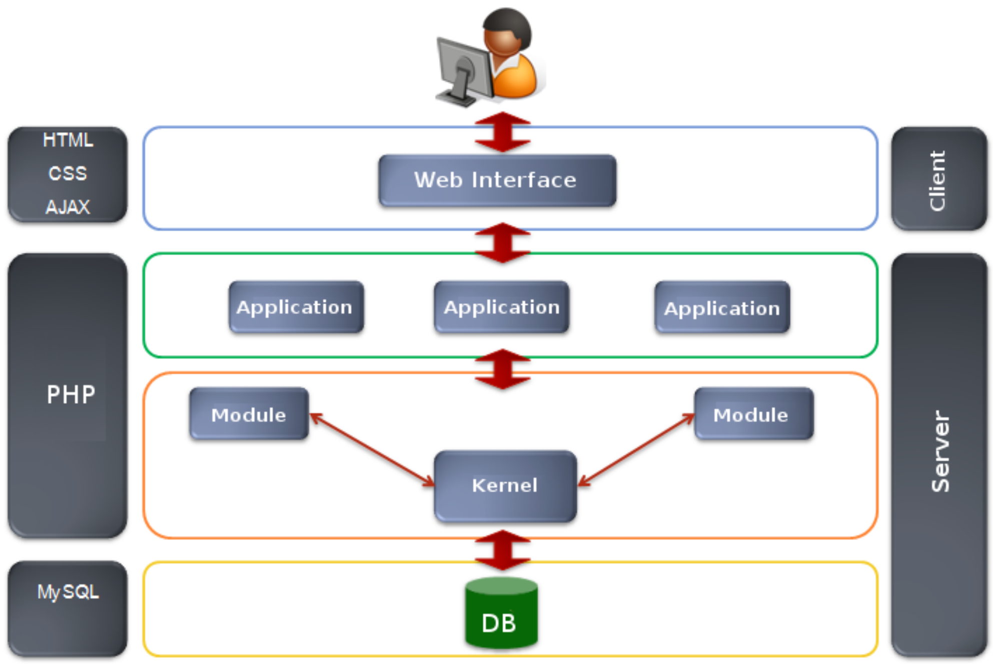
The controller is responsible to handle the other two models working together. This module receives a request from a client, invokes the *model* to perform the requested operations and send the resulting data to the *view*. The view is in charged to format the resulting data to be presented to the user. This format is in HTML format.

4

Architectural and Component-level Design

4.1 System Structure

4.1.1 Architecture Diagram



Caption : Block Diagram showing major components.

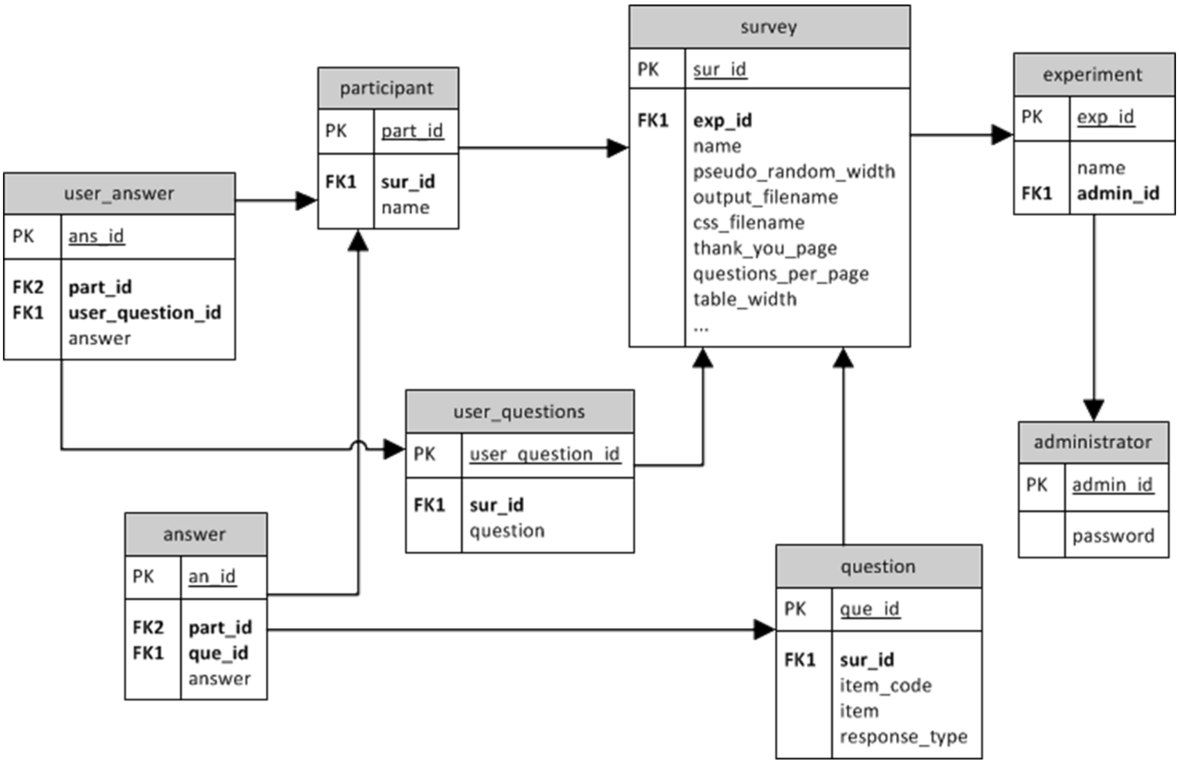
The figure shows the *Architectural Diagram*. This diagram shows the main components in our application. The main characteristic the application will be divided in different levels of interaction.

The database will only interact with the *Modules*, that will be responsible for retrieving and storing data from the database.

The *Application* will be responsible for controlling the interaction between the user and the database. Through the *Web Interface* the user will request information that the application will receive invoking the correct module.

The tools used to develop the System are HTML, CSS, PHP and SQLite.

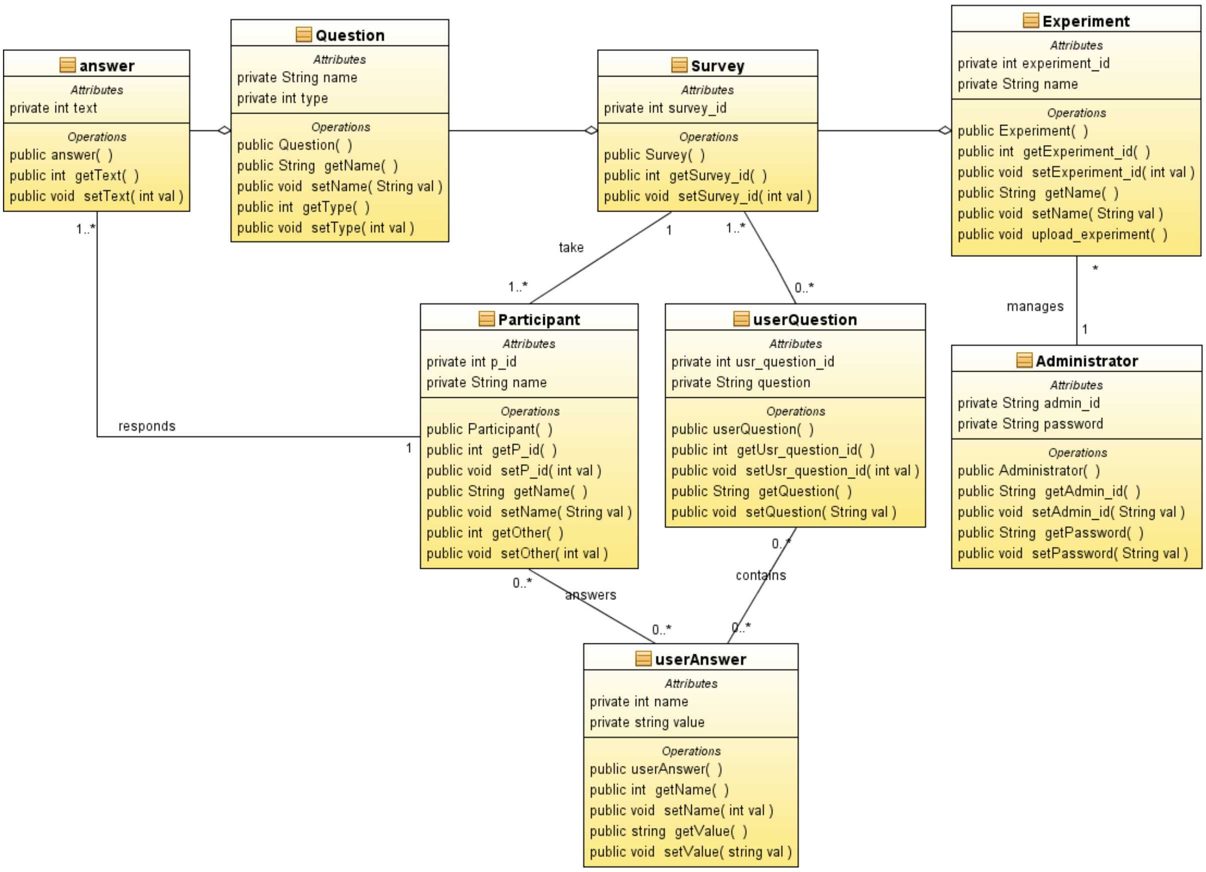
4.2 Database Diagram



Caption : Database Diagram.

This diagram shows our database. It shows the main interaction between tables and their attributes.

4.3 Class Diagram



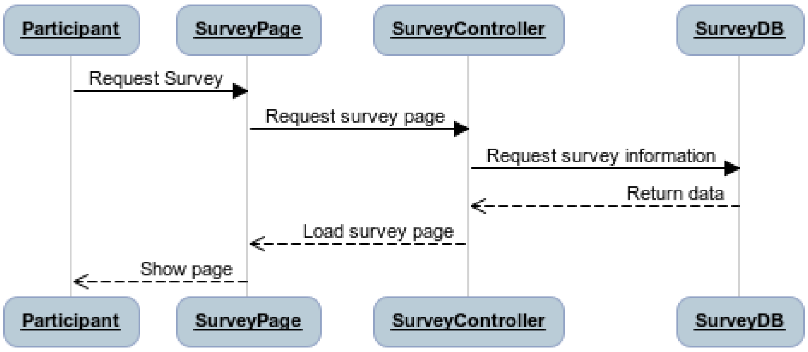
Caption : Class Diagram.

This diagram describes the structure of the survey management system by showing the system's classes, their attributes, and the relationships between the classes.

4.4 Sequence Diagrams

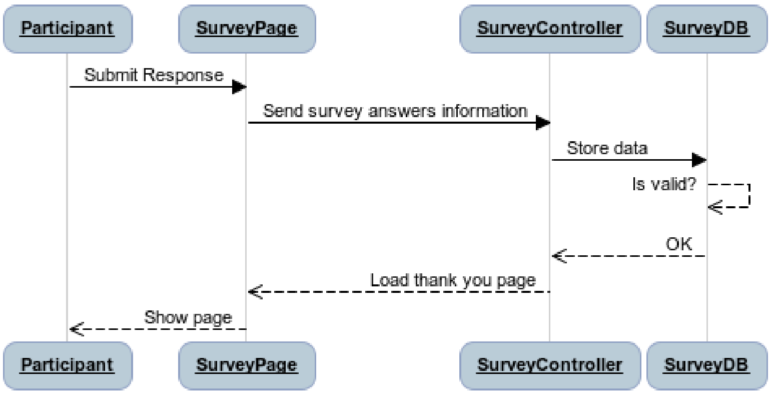
These interaction diagrams show the order of operation of the different processes or objects in our system. The interact with each other by sending messages (represented by the horizontal arrows).

4.4.1 Request Survey



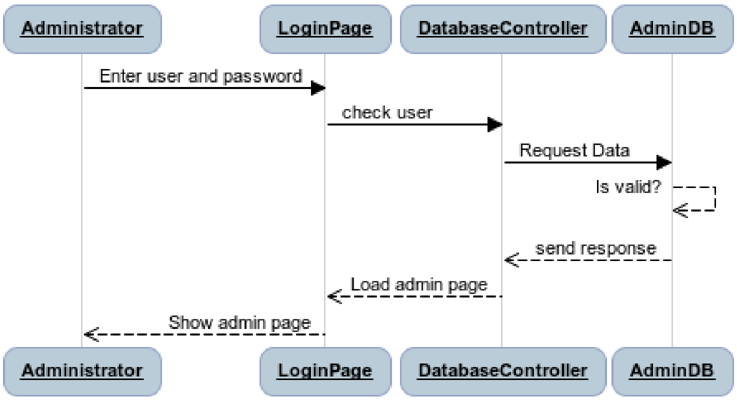
Caption : Request Suvey Sequence Diagram.

4.4.2 Submit Survey



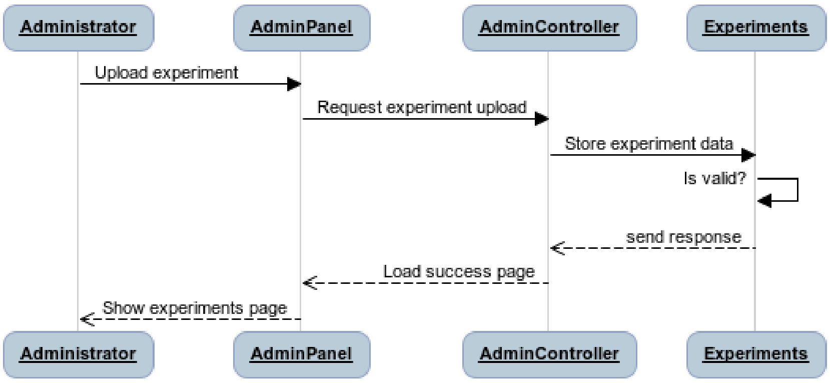
Caption : Submit Survey Sequence Diagram.

4.4.3 Login



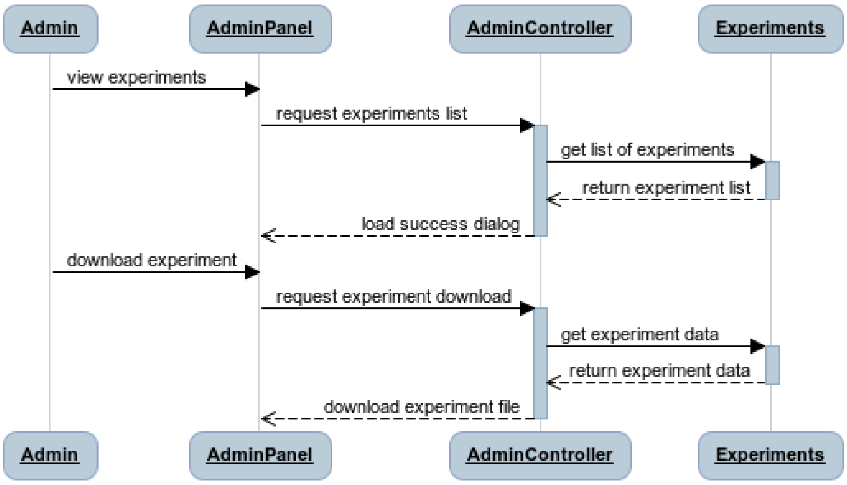
Caption : Administrator's Login.

4.4.4 Upload Experiments



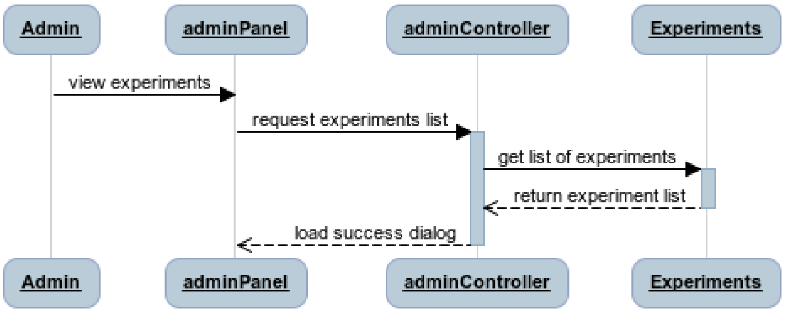
Caption : Upload experiment.

4.4.5 Download Experiments



Caption : Download Experiment.

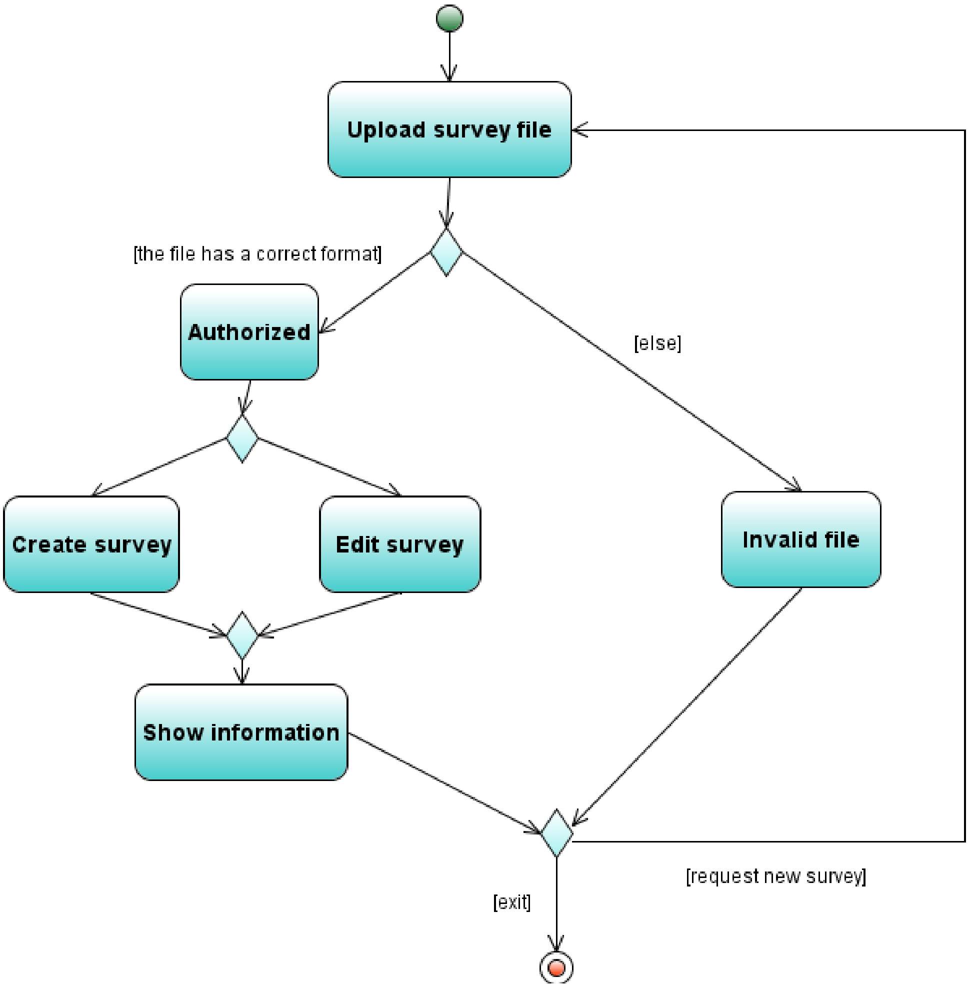
4.4.6 View Experiments



Caption : View Experiments.

4.5 Activity Diagrams

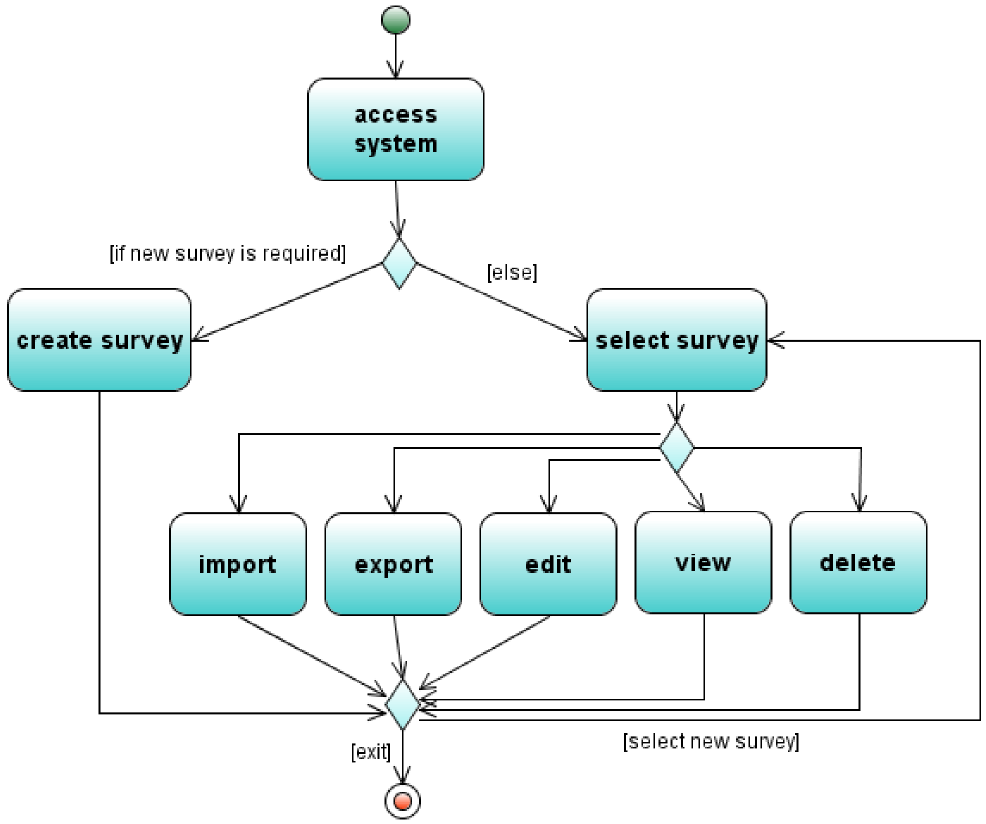
4.5.1 Survey creation



Caption : Activity Diagram for survey creation.

This is a graphical representation of workflows of the activity of survey creation. It shows the steps needed to be followed in order to upload a new survey (or as we call it *experiment*). The administrator selects a file to upload. They system will check the validity of the file format, if correct the user can either make final changes or create it. If the format is incorrect, the system will show the error and request a new file.

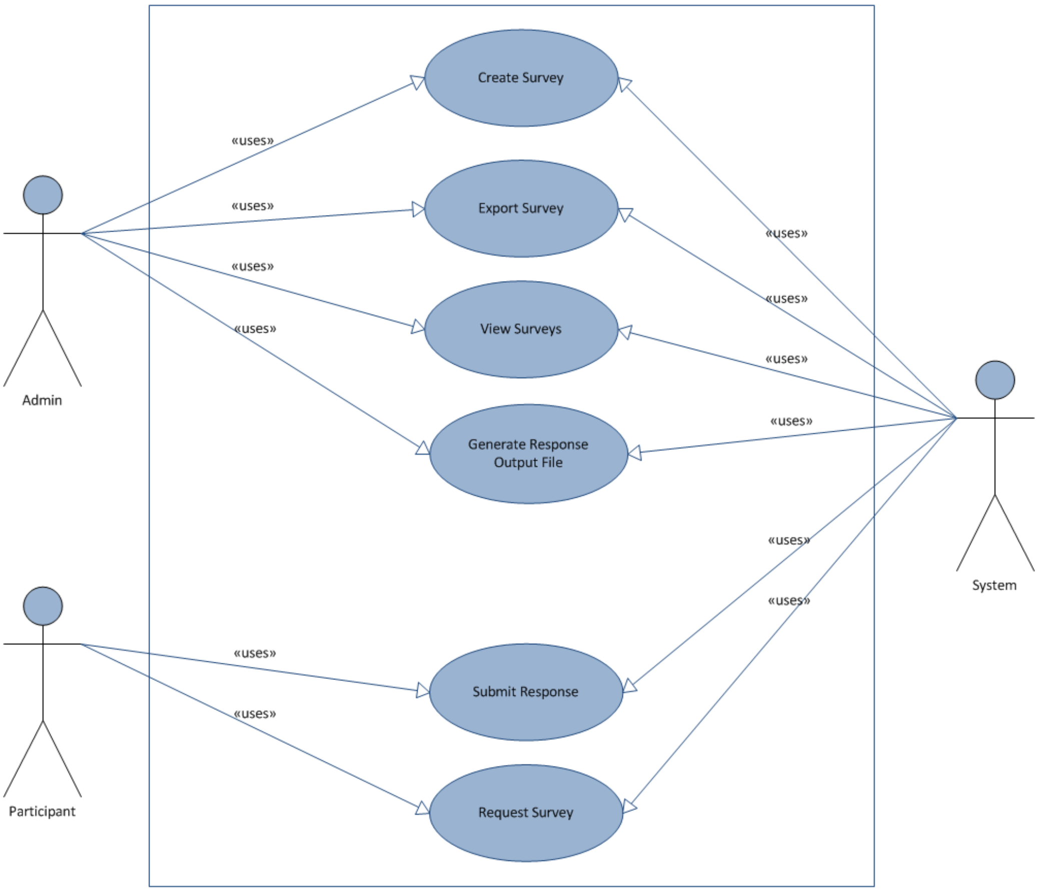
4.5.2 Survey Management



Caption : Activity Diagram for survey management.

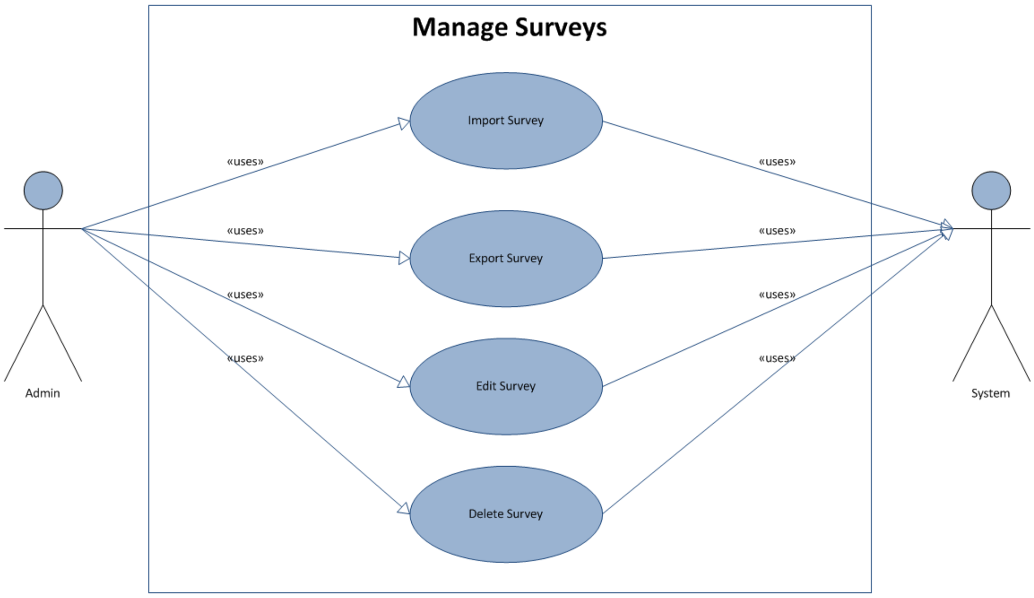
This activity diagram represents operational step-by-step work-flows of the survey management. The main options that the system will allow the administrator to perform related to experiments is their creation or edition. In order to perform basic operations such as *import, export* or *delete*, you need to select the survey to which you are going to apply the operations.

4.6 Use Case Diagrams



Caption : Main Use Case Diagram.

This use case shows the main basic functions of our system. There are to main actors that will interact with the system, the *administrator* and the *paticipant*. Each of them will be able to perform some tasks, as specified by the previous figure.



Caption : Survey Management.

This use case represents the basic interaction of the administrator with the Surveys. The surveys is the main focus of our application, that is why the Administrator has capabilities for total interaction with them.

5

Restrictions, Limitations, and Contraints

5.1 Description For Experiment Component

An experiment will consist in a set of surveys. Each survey should be able to contain three types of inputs:

Text box. It will consists of 0 or more integers.

Radio box. Containing one number. It is inclusive.

Yes/No radio box. Allowing a two choice answer.

The system will support infinite surveys and experiments.

5.2 Description For User Component

The user or 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

5.3 Description For Administrator Component

Although our design will allow the possibility of future additions of administers easily, initially we will only have one. 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

5.4 Description for the Output File

The exported results will be in **csv** format. The mush contain the following restrictions:

- Additional column for survey title.
- Do not have to support excel.

6
User Interface Design

6.1 Description Of The User Interface

The main features that the software will present are:

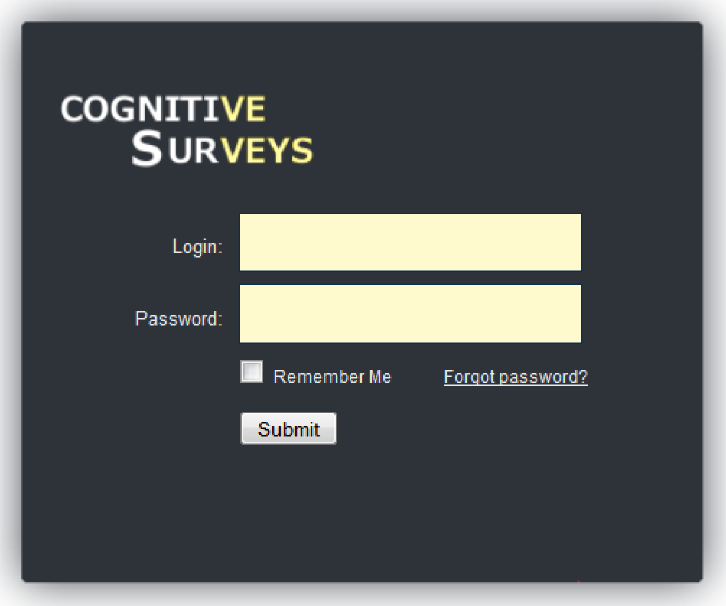
- *Sets of surveys*, allowing the user to automatically loop through them.
- *Pseudo-Randomization* of the order of the surveys.
- *Result file.* This file will contain these main fields: Subject ID, survey number, code and responses.

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

6.1.1 Screen Images

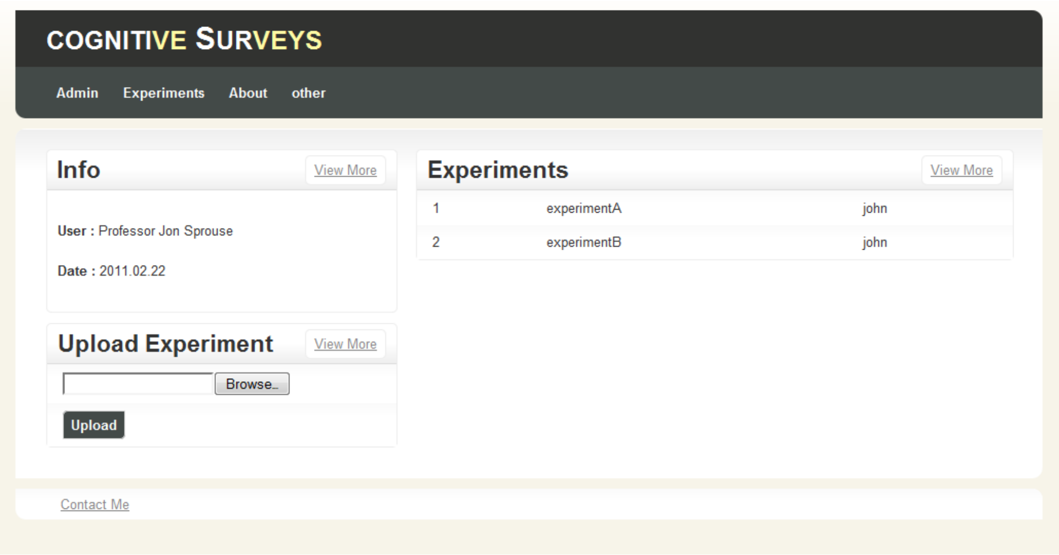
Login Screen



Caption : Login Screen.

This is the first page that appears to the administrator while the application. It will request an user name and the password. If the input is correct (matches with the records in the database), the main administration panel will show up.

Admin Screen



User : Professor Jon Sprouse

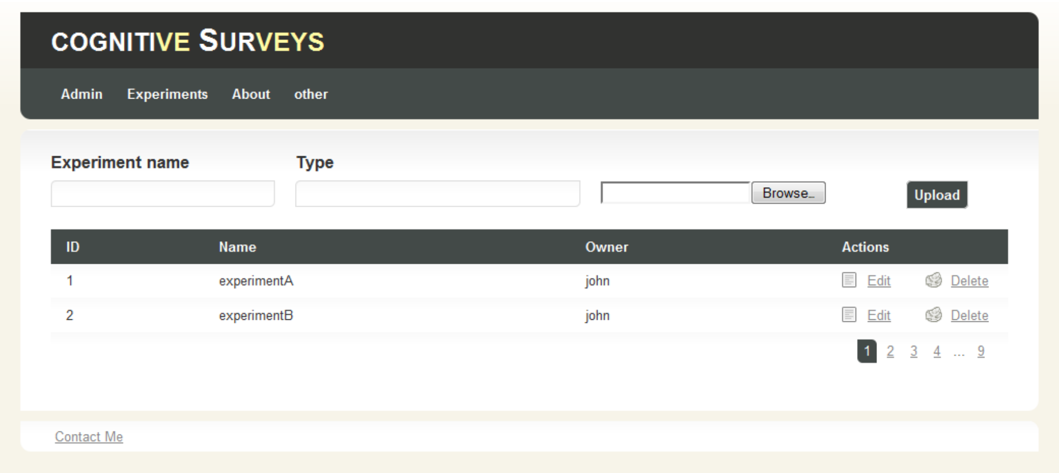
Date : 2011.02.22

1	experimentA	john
2	experimentB	john

Caption : Adminisrator Screen.

Main page where the administrator can perform the main operations of survey management.

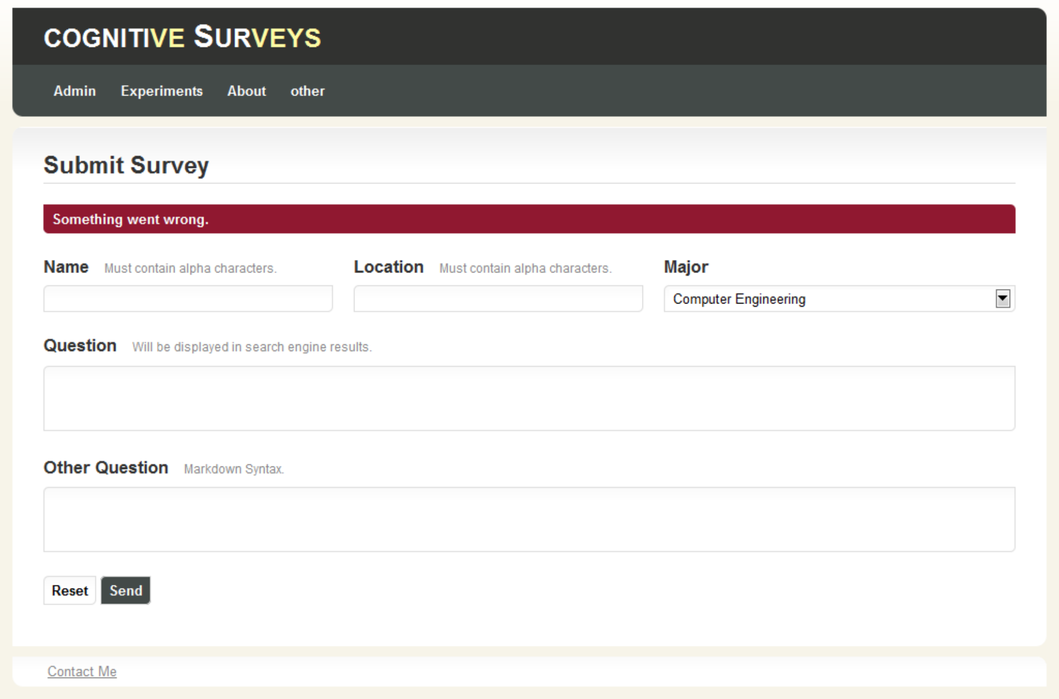
List of Experiments Screen



ID	Name	Owner	Actions	
1	experimentA	john	Edit	Delete
2	experimentB	john	Edit	Delete

Caption : Screen showing list of experiments.

Survey Screen



Submit Survey

Something went wrong.

Name Must contain alpha characters.

Location Must contain alpha characters.

Major: Computer Engineering

Question Will be displayed in search engine results.

Other Question Markdown Syntax.

Caption : Survey Screen.

Sample screen of how a survey might look like. The red high-lighted text shows when the user sent the survey but he did not fill all fields correctly.