Web Technologies Report

Pair: yw17489 jl17471

Lu&Yue Photo Gallery:

The topic of the website is "Travel Album". It would be a website for users to register an account, login to upload photos, view photos, and leave comments and so on.

User menu:

Execute *npm install* command in the root directory, which is install the node modules automatically.
Node *index.js* in the root directory
Open a brower (Best for Chrome) and access http://localhost:8080

Estimate marks:

A for HTML
A for CSS
A for JS
A for PNG
A for SVG
A for Server
B for Database
A for Dynamic pages
B for Depth (out of 40)

Implementation detail:

HTML:
We used the XHTML approach with occasional validation.
We have known the structure of the HTML pages. We generated all the html by ourselves and used several elements of the HTML.
We use **Handlebars** view engine to generate HTML pages.

CSS:
We wrote style.css and form.css by ourselves and gained some experience with things like the decoration of the website layout, the beautify of forms and the hover effect of buttons.

We produced a navigation bar via **bootstrap** framework.

We used the framework from http://daneden.me/animate to generate the animation effect.

We also use the framework from resources to generate the grid effect of photos on home page and the scroll effect of album.

JS:

We wrote the function about validation for form, get cookie and display on the website by ourselves.

We produced the time out function from resources.

We produced scroll animation for album using jQuery framework http://www.smoothdivscroll.com/.

PNG:

We design the bitmap graphic background by **GIMP,** add the logo svg file in the background.

Figure 1 background

Basic techniques:

Convert images export to PNG, crop edges and change resolutions.

Use select tool and myPaint brush tool to remove the original text.

Use color picker tool and bucket fill tool to change the colors.

Advanced techniques:

Flexibly use the tools such as scale, alignment and text.

Flexibly **handle layers** and transparency, such as merge down and add Alpha channel.

Use the free select tool to cutout image and combining the existing layers.

SVG:

Design the vector graphic logo of our website, the inspiration comes from internet and design it by **Inkscape**. This is a cute penguin names "Lucky" with a camera in her hand, and she invites users to share their photos on our website.

Figure 2 penguin logo
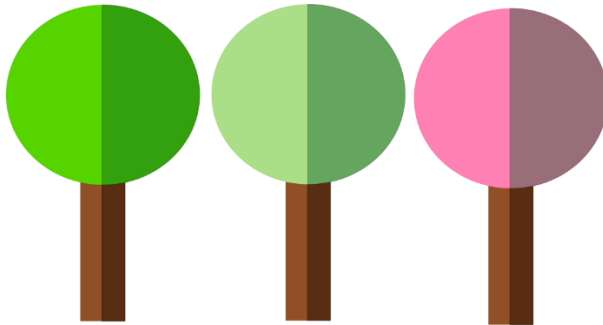

Figure 3 photo-yue

Figure 4 photo-lu

Figure 5 icon

Figure 6 camera

Basic techniques:

Draw simple shapes, freehand drawing, fill colors and adjust shape and transparency (Figure 6).

Draw Bezier curves and **edit paths** by nodes (Figure 4).

Draw the penguin's smile eye by using calligraphic or brush strokes (Figure 2).

Edit document properties to adapt the size of image.

Advanced techniques:

Apply put **Text on Path**, let text "Lu&Yue Photo Gallery" displayed follow a curve (Figure 2).

Scale rounded corners, incision or change radian of basic shapes, such as rectangle, circle and star (Figure 2,5).

Trace Bitmap and Object Ungroup to design the Figure 3.

Edit objects by grouping, rotating and aligning (Figure 5).

Server:

We use the express module in Node js to build the website.

We restrict unsafe url and use valid-url module to **validate** the url before get and post function.

We use cookie to store the user name of the website and clear the **cookie** when logging out.

We use the module multer to **upload photos** from client side to the directory *uploads*.

Basic compatibility test is tested on Chrome and IE 11.

Database:

We use **sqlite3** to store data and do queries. We consider the **SQL injection** problem and use serialize method to call functions.

The database is simple but all the tables are based on **BCNF**.

We have managed to insert data as well as extract it. We use some queries on sever side to keep consistency. We also use queries to check the data which to be inserted to avoid the error thrown by database. Besides, the password will be **encrypted** by hash algorithm MD5 before inserting into database.

Dynamic pages:
We produce dynamic pages by inserting data into templates on the server side. We use **Handlebars** as view engine and use **hbs** module to create pages. The "result.hbs" would help to deliver all kinds of result information after operation on website and the home page will show 3 records of table comments in database.

Depth:
We design this website to upload and share photos. We have managed to make the website look clear and nice, and we searched for the cool bootstrap and jQuery framework for photo gallery effect and used them in our website.

We have managed to create database for accounts of the website and user can log in and log out and store username into cookies. We also added the uploading files function. We deliver all the pages dynamically using view engine tool and this can make our website look consistent via layout template.

We make sure that under each heading we have investigated different kinds of issues and written a substantial code ourselves. In addition, we also learn to look for and use framework from resources and adapt it fluently in our website.

We basically use the print sentences in console to test and debug and it worked well.

We made a lot of efforts to deal with problems about cookie. We used the cookie to store user name but when requesting cookie on server side it always shows undefined, then we solved this problem by getting cookie from client side. Finally we realize maybe using session would be a better choice but we did not try that.

Frameworks and Libraries

- Node modules
  - express
  - express-handlebars
  - sqlite3
  - body-parser
  - cookie-parser
  - hbs
  - crypto
  - url
  - valid-url

- ➢ multer
- Bootstrap
- GoogleFonts
- jQuery