Assignment 1: Rubric

Important: This rubric is a guideline for marking and should not be considered as binding. It means that the course staff reserves the right to:

- modify the rubric after the submission deadline
- assign more or less weight to each rubric elements when it comes to give a score

Complete this rubric by filling the blanks and selecting options using an x whenever it is appropriate. Here is an example that you should modify as well:

1. My name is Joshua and this rubric is [select one]:
    - _ missing
    - _ incomplete for the most part
    - _ badly filled for several questions
    - _ good but couple of questions were not filled or wrongly filled
    - x perfect

Academic integrity declaration

I hereby declared that the work submitted here is mine and its production complies with

1. the Code of Behaviour on Academic Matters of University of Toronto
2. and the course policy (see website)

[date] 01/27/2017

[signature or initials] JG

Designing the UI

Grade (TA only)

- max1: 40
- score1:

1. The UI elements for the **uploader form** are: [select one]
   - ○ _ missing
   - ○ _ almost missing with couple of exceptions
   - ○ _ mostly there with couple of exceptions
   - ○ x all there without any exception
2. The UI elements for the **display** are: [select one]
   - ○ _ missing
   - ○ _ almost missing with couple of exceptions
   - ○ _ mostly there with couple of exceptions
   - ○ x all there without any exception
3. The UI elements for the **the comment form** are: [select one]
   - ○ _ missing
   - ○ _ almost missing with couple of exceptions
   - ○ _ mostly there with couple of exceptions
   - ○ x all there without any exception
4. The UI elements for the **the comment section** are: [select one]
   - ○ _ missing
   - ○ _ almost missing with couple of exceptions
   - ○ _ mostly there with couple of exceptions
   - ○ x all there without any exception
5. The CSS layout (not the code) is: [select one]
   - ○ _ is almost inexistent (close to the default layout)
   - ○ _ there but chaotic: elements are either misplaced, not aligned, or overlapping each other)
   - ○ _ ok but several elements were neglected
   - ○ _ good but couple of elements were neglected
   - ○ x clean and organized
   - ○ _ beyond expectations
6. The HTML code [select many]:
   - ○ x contains the required elements: head, meta charset, title, body
   - ○ x use only relative paths
   - ○ x use the `<img>` tag for data only (not UI)
   - ○ x enforces a clear seperation between the content (HTML), the style (CSS) and the processing (Javascript)
   - ○ x makes a correct use of id and classes attributes
   - ○ x does not use deprecated elements
7. The CSS source code [select many]:
   - ○ x use only relative paths
   - ○ x use CSS background properties for UI elements (not data)
   - ○ x does not have a large number of shadowed properties
8. All design components borrowed from the web are credited in `credit.html`: [select one]
   - ○ _ no at all
   - ○ _ some of them but many are missing
   - ○ _ most of them but couple are missing

- x all of them without any exception
9. Overall, the quality of the UI is: [select one]
   - _ not implemented
   - _ poor
   - _ fair
   - _ good
   - x excellent
   - _ beyond expectations

In the box below, write any comment you would like to communicate the TA regarding the work done for this part:

Comments

```
The UI is clean and no elements get covered.
```

Building the features

Grade (TA only)

- max2: 40
- score2:

1. The application: [select many]
   - x does not use any server side feature
   - x does not use any library
2. The **image uploader** as described in the handout: [select one]
   - _ is unimplemented
   - _ is somehow implemented but does not realized its main functionality overall
   - _ does achieve its main functionality overall but several features are missing
   - _ does work well but couple of features are missing or not working properly
   - x does work very well overall and all features are properly implemented
3. In the box below, write down the features that you were not able to implement or make it work properly:
   Comments

```
Image uploader checks for errors by not submiting if part of the form (title,
author, file) is empty. Comment form checks for errors by not submiting if part of
the form (author, message) is empty.
```

1. The **image browsing** as described in the handout: [select one]
   - _ is unimplemented

- ○ _ is somehow implemented but does not realized its main functionality overall
- ○ _ does achieve its main functionality overall but several features are missing
- ○ _ does work well but couple of features are missing or not working properly
- ○ x does work very well overall and all features are properly implemented

2. In the box below, write down the features that you were not able to implement or make it work properly:
   Comments

```
Images have a max size so they do not cover other elements on the page. The size of
the image area is the same for all images so the screen does not bounce around when
navigating through the different images. The navigation arrows extend to the entire
height of the image so you do not have to click directly on the arrow.
```

1. The **comments** as described in the handout: [select one]
   - ○ _ is unimplemented
   - ○ _ is somehow implemented but does not realized its main functionality overall
   - ○ _ does achieve its main functionality overall but several features are missing
   - ○ x does work well but couple of features are missing or not working properly
   - ○ _ does work very well overall and all features are properly implemented

2. In the box below, write down the features that you were not able to implement properly:
   Comments

```
When you delete a comment, it gets deleted and the 10 most recent comments are
always shown. Could not get it to stay and display the section of the comments that
the comment was deleted from.
```

1. The **navigation** as described in the handout: [select one]
   - ○ _ is unimplemented
   - ○ _ is somehow implemented but does not realized its main functionality overall
   - ○ _ does achieve its main functionality overall but several features are missing
   - ○ _ does work well but couple of features are missing or not working properly
   - ○ x does work very well overall and all features are properly implemented

2. In the box below, write down the features that you were not able to implement properly:
   Comments

```
When an id is given in the url that is wrong, the user is sent to a seperate 404
page.
```

1. Overall, when using the application: [select one]
   - ○ _ it does not work because several features are not implemented
   - ○ _ it does not work well as a simple usage does not give the expected results or generates errors
   - ○ _ it works relatively well but a heavy usage does not give the expected results or generates errors
   - ○ x it works really well under all kinds of usages
   - ○ _ it works exceptionally well and tolerates bad usages (security aside): bad user inputs, UI stress and so on

In the box below, write any comment you would like to communicate the TA regarding the work done for this part:

Comments

```
File upload button only allows users to upload images, but url upload does not
check to make sure if the given url is an image.
```

Code quality and organization

Grade (TA only)

- max3: 40
- score3:

---

1. The repository is overall: [select one]
   - _ not following the required structure
   - _ follows the required structure but some files are either misplaced, wrongly named or misspelled
   - x well organized
2. The HTML code: [select many]
   - x is clean, well organized and indented properly
   - x does not contain any error from the w3c validator
   - x does not contain any warning from the w3c validator
3. The CSS code: [select many]
   - x does not repeat itself (DRY principle) and is easily maintainable
   - x is clean, well organized and indented properly
   - x does not contain any error from the w3c validator
   - x does not contain any warning from the w3c validator
4. The Javascript code: [select many]
   - x is clean, well organized and indented properly
   - x does not contain any error from JSHint
   - x does not contain any warning from JSHint
   - x does not repeat itself (DRY principle) and is easily maintainable
5. When the app executes: [select many]
   - x it does not generate superfluous debugging messages in the console
   - x it does not generate error messages in the console
6. Overall, the application code: [select one]
   - _ is missing for the most part
   - _ is poorly implemented
   - _ is fairly well implemented

- ○ _ is good
  - ○ x is excellent
  - ○ _ is beyond expectations

In the box below, write any comment you would like to communicate the TA regarding the work done for this part:

Comments

```
your comments go here.
```

Neatness

Grade (TA only)

- max4: 20
- score4:

---

1. What are the things that be considered as "neat": [select many]
   - ○ _ the outstanding CSS design
   - ○ x one or several improved features (please describe the(se) feature(s) in the comment box below)
   - ○ _ one or several extra features (please describe the(se) feature(s) in the comment box below)
   - ○ _ something else (please describe in the comment box below)

In the box below, write any comment you would like to communicate the TA regarding the work done for this part:

Comments

```
If there are no more images to the left or right, the arrows disapear to show that
there are no more images in that direction. If there are no more newer or older
comments the newer or older buttons will disapear to show that there are no more
newer or older comments. If a comment becomes too long a scroll bar appears to
allow users to scroll through the whole comment.
```