Exit Report – Website Design

|  |  |
| --- | --- |
| Name | Jamie Ahloy Dallaire |
| Contact email: | j.ahloy.dallaire@gmail.com |
| Task/Role | Website design |

Just leave a section blank if you don’t feel the question applies to your task.

|  |
| --- |
| Describe your role/task: Creating and managing the conference website. |

|  |  |
| --- | --- |
| What did you wish you had known beforehand? | List important info or decisions you may have to know in order to complete your task/role. For example, you can’t start fundraising until you have a budget.   * Check whether the uni’s/department’s servers support PHP or some other scripting language, then test whether it works, before even trying to set up a registration page. |

|  |  |
| --- | --- |
| Contacts | List the organization/companies/individuals you contacted to complete your task. Would you contact those individuals again? If not, why and do you have suggestions for may have been a better choice?   * Guelph Computing & Communications Services (<https://www.uoguelph.ca/ccs/>) who host the website. They provide a fine basic service, but have limited options regarding support for scripting languages, and they were only moderately helpful about it. |

|  |  |
| --- | --- |
| What worked? | List the things that made your task easier, or resulted in it being a success.   * Communication between the person checking the email and myself. I would get updates about website issues and address them immediately. * Our registration form and payment forms were separate, and initially some payments came in from people who had not registered themselves (they were paying for others). Adding a ‘name of registrant’ field to the payment form solved this. * Google Forms works great for registration, and automatically feeds entered data into an online spreadsheet. The spreadsheet can be edited directly or as an offline copy to keep track of e.g. who has paid or not. * Saving a custom Google Map with pins and short blurb on important conference locations worked well. * Kompozer is a good basic open-source html editor I used to make the page. I tried several programs and ended up sticking with this one (it’s a straightforward what-you-see-is-what-you-get editor). Best if you do have some degree of familiarity with html, to be able to sort out the kinks manually when they happen. * It was very beneficial to take a few hours to look up principles of web usability, appealing design, and accessibility before starting. * Test early and often, on multiple devices and browsers. Get committee members and/or loved ones to give the website a look when you make major changes. We caught a bug this way that messed up display specifically on mobile Apple devices running Safari, which I never would have caught myself. * When registration ended, we didn’t disable the registration page, but instead moved it to a different (secret) address. This let the email person give out the link on a case-by-case basis, e.g. to late registrants or to people who’d registered on time but had info missing. |

|  |  |
| --- | --- |
| What didn’t work? | List the things that you would change or do differently to save yourself time/energy/$ if you had to do the task again.   * Wasted much time trying to set up a PHP registration page before learning that despite the servers running PHP, they did not actually support it for organizational web pages. Similar story with perl, which never ended up working either. Had I known, I would have started with Google Forms. * If possible (e.g. on a server that supports scripts), don’t decouple the registration and payment forms (i.e. make people pay when they register, leaving an option to defer payment for exceptional circumstances). Having those two separate just made for headaches and for the person responsible for the email account having to hound people for money.’ * Be aware that certain questions on the registration form may not elicit reliable answers from registrants. For example, nearly 100 people responded that they did plan to attend the opening evening cocktail and plenary, but in fact only about half that many showed up. So we bought too much food. Perhaps the way these questions are formed can remediate this; otherwise be prepared for some overestimation. |

|  |  |
| --- | --- |
| Give a general timeline to completing the task. | Give general dates for around when milestones on your task were completed. You can make suggestions about if you thought you did this too late/too early.   1. November: look into web hosting services offered through the university or department. 2. Mid January: create offline prototypes to get committee feedback. 3. Mid January (too late): sign up for web hosting service (in practice this took a very long time with paperwork and approvals, so do it earlier – it is surely very quick if you pay for a commercial server instead). 4. Late January: Put up website, without active registration page. 5. Ongoing: Update website, as necessary 6. Late February: Put up registration page, then monitor and deal with any issues. 7. Early April: Take registration page offline (see above: in practice it was simply moved/hidden). 8. Late April: Post conference program and update with any last important details (e.g. parking info, presentation uploading instructions). 9. Late May: Following the conference, post a final update proclaiming the conference a success and adding e.g. award winners, next year’s conference location, pictures, etc. |

|  |  |
| --- | --- |
| Do you have any online accounts or account information to pass along? | Give id and passwords for account you may have used and the next group could take over (Facebook, Twitter, etc.).  None. |

|  |  |
| --- | --- |
| Future Goals | If you could do this task over again, what do you think are reasonable goals that would be achievable?  No comment |

|  |  |
| --- | --- |
| Other comments | List any other useful information that the previous questions may not have answered.  General Description   * Website files are in the Dropbox folder. Western’s 2013 website is also there, and contains some useful scripts that can be modified for registration if your server supports them. * Home page is interactive with a Twitter feed. If viewing the context off the server, this will not display correctly until on a server.   Registration page   * Used Google Forms, but have reproduced comments regarding Western’s 2013 registration page in case useful: * *Written in PHP.* * *If viewing the context off the server, the page will not display correctly (you will see a lot of code) until on a server.* * *Creates an excel file (.csv) of all entered information. Saved a lot of time versus getting a whole bunch of emails for abstracts. You can also create a document of formatted abstracts, see below.* * *To generate formatted abstracts (see registration.xls in oe3c\_web for example): insert columns of <p> HTML code to create new paragraphs between information on a separate sheet. Save the sheet as tab delimited .txt. Change the extension from .txt to .htm. Click on file to see formatted abstracts.* |