Evaluation of Overleaf Based on Nielsen's 10 Usability Heuristics Report

170011474

October 2019

Abstract

Nielsen's 10 usability heuristics for user interface design has been adopted to evaluate interfaces by many research groups, showing a strong capability of a complete set of heuristics or guidelines in evaluation phases. This set of heuristics can return feedback to developers in the early phase of design, including positive features and drawbacks. Therefore, adjustments can be made and a better product design can be achieved. This report aims to evaluate Overleaf based on Nielsen's 10 usability heuristics individually and also a group evaluation will be conducted. A top list of positive features and negative issues with brief justification and potential improvement or solutions to them will be presented.

1 Introduction

Overleaf, an online LaTeX editor, provides users with real-time previews and version control, hence users can deliver a professional scientific paper within one integrated platform. It also allows multiple users to collaborate with other authors concurrently and to access to the latest version of a document from any client devices. Although it is widely acknowledged that Overleaf accelerates the process of publishing a formal article and offers users an aesthetic design, it still has some problems and drawbacks to be discussed.

2 First Evaluation Phase

The operating system adopted to inspect Overleaf is Linux Ubuntu 18.04, with Google Chrome browser installed. The Overleaf account was signed by using my Gmail, and a template paper of Human Computer Interaction (HCI) conference was used (see CHI2020-Proceedings).

The inspection was taken systematically based on Nielsen's ten usability heuristics (Nielsen, 1995). Firstly, for each heuristic, I read its definition and checked if Overleaf satisfied or violated it. Then, after going through each heuristic, I wrote down my mistakes or wrong clicks during my inspection process. The next step was to find out reasons why I made mistakes: was there any confusing buttons or ambiguous instructions in Overleaf? Did they violate any heuristic? Therefore, by bilateral inferences: find advantages and disadvantages based on each heuristic; find corresponding heuristics based on my mistakes, I developed a complete table containing positive features and negative ones.

After identifying those features, justifications for them were made. In order to be more familiar with Overleaf and collect data, I used Overleaf to write a report for CS3099 assignment with my team members. I observed and recorded their behaviour and use those results to verify my hypothesises.

3 Top 5 Positive Features and Top 10 Issues Identified

Notice that total number of Agreement is 4, including the reporter herself.

Top List for identified Features and Issues **Individual Summary Table**				
Order	Top 5 Positive Features	Agreement	Top 10 Issues	Agreement

1	A warning dialog and a reconnecting dialog popped up when losing connection to online platform	2	Placement of Menu and Back buttons on the top left leads to wrong clicking	3
2	Red error triangles and yellow warning triangles appearing in the left side bar	2	No undo and redo buttons	2
3	Universally recognisable icons and buttons	4	Users need to remember if the file is modified since they last compiled the document	1
4	Multi-user mode shows individual cursors' positions in different colours (follows collaboration standard)	3	Some error messages are ambiguous and confusing	4
5	Adequate helping documentation	2	The leftward and rightward arrow buttons located between the source file and PDF file are misleading: users easily regard them as full screen buttons	1
6			A wrong type would not invoke an error triangle until compilation (For example, \textbfffff)	1
7			Users directly log out after an accidental clicking "log out" without a confirmation box popped out	1
8			No shortcut to jump back to the last cursor position	1
9			No auto formatting option to indent lines based on writing styles (Visual Studio Code uses Ctrl + Shift + I in Ubuntu to format document)	1
10			Some hotkeys are not externally consistent (Ctrl + U to obtain upper case while Microsoft Word uses it to underline selected sentences)	4

The reasons why these 5 positive features are more important than the rest:

1. Heuristic 1 **"Visibility of system status"** represents that users are able to be notified about the current process given an appropriate feedback. A warning dialog showing the Internet connection is necessary. Otherwise, if the Internet connection fails, unavailing efforts are made by users, which reduces the writing efficiency. Therefore, a warning box for Internet status is a premise of the subsequent manipulation.

2. Heuristic 9 **"Help users recognise, diagnose, and recover from errors"** requires the system to provide opportune messages for users after errors occur. Those visible red and yellow triangles in the side bar can notify users which specific lines should be debugged, significantly enhancing efficiency to recognise and diagnose errors.

3. Heuristic 2 **"Match between system and the real world"** states that the system should speak the users' language and those instructions should be familiar to users. Users can understand buttons and icons intuitively and quickly, which saves time for them and also reduces the probability of making mistakes.

4. Heuristic 1 **”Visibility of system status”** is satisfied since concurrent status will be shown on the top in multi-user mode, and also individual cursors in different colours can distinguish one author from other authors. Therefore, users easily figure out the current status of each author.

5. Heuristic 10 **”Help and Documentation”** illustrates that help and documents are conducive for the completion. Adequate documents and corresponding explaining videos in Overleaf undoubtedly meet this heuristic, hence users can easily search for their desired function. Those helper documents are divided into two parts: one for the usage of the common tags for layout and formatting, such as table insertion and image insertion, and the other one for suggested solutions to common errors users may encounter.

The reasons why these 10 issues are more important than the rest:

1. The order of Menu button and Home button is misleading, which breaks the heuristic **”Consistency and standards”** since users easily click the Menu button for returning back to the project home page. On other platforms, however, developers put the home page button at the very top left position for which users are often more familiar with this arrangement.

2. No redo and undo buttons visible for novice users violates the heuristic **”Flexibility and efficiency of use”** as well as **”User control and freedom”**. Users need to take time to search for a helping suggestion online and to remember corresponding shortcuts, and this is the only way to undo and redo. Thus, this problem brings lower flexibility and lower efficiency.

3. The ”Recompile” button on the top of the preview page enable users to recompile the current document. However, users need to recall whether they have modified the newest version or not, and they tend to recompile it again even if they do not modify it. This issue would not cause disastrous crash but the compiling time will increase if the document is large. This problem reflects a violation of the first heuristic **”Visibility of System Status”** because users cannot know the current status of compiling. It also violates the heuristic ”Recognition rather than recall” since users need to remember if the compiled PDF file is updated.

4. Ambiguous and disorganised error messages violates the heuristic **”Help users recognise, diagnose and recover from errors”** and **”Help and documentation”**. Sometimes the error message can show up if the mouse is hovering over the yellow triangle, but the message will overflow the current window so users cannot see the whole message. Thus, diagnosing and debugging a file becomes more troublesome and time-consuming.

5. Misleading arrow buttons in the middle of source file and preview PDF file violates the heuristic **”Consistency and standards”**. Most users easily have intuitions that the rightward and leftward arrows will bring a full screen mode of source file or PDF file respectively, while they simply providing ”going to the code/PDF location in PDF/code” with users. Thus, this problem confuses users.

6. Typing a wrong tag name would not report warnings until users find it on the compiled PDF. This problem goes against the heuristic **”Error prevention”** which stands for eliminating error-prone conditions at the first place before users commit. When it comes to some mathematical equations, the situation worsens since any inappropriate tag or parentheses will lead a truncating or omission. Those problems are not prevented in the first place.

7. Heuristic 5 **”Error prevention”** is violated, as an accidental clicking on ”log out” will directly lead to logging out without a confirmation box popped out to users. Therefore, consequences of accidental clicks on ”logging out” cannot be prevented in the first time, and users need to waste more time on things irrelevant to their works.

8. After users wheel up to modify the previous part, they probably encounter a problem that they cannot backtrack to the last cursor position in a short time due to there is no hint where the cursor used to be in a large file. Thus, this violates the heuristic **”Recognition rather than recall”** as users need to remember the cursor position.

9. The lack of an automatic formatting tool results in a messy format in the editing page if users are loath to indent their texts properly. Although this would not affect the format of documents eventually, it still rises issues for users to retrieve information when they are trying to track back to the previous section. Particularly speaking, if multiple tables or mathematical equations are inserted into the document, scarce indentations would bring them a nightmare. This problem violates the heuristic **"Match between system and the real world"** as the messy presentation is hardly seen in the real world, and also violates the heuristic **"aesthetic and minimalist design"**.

10. Hot keys are very inconsistent with other hot keys from other platforms, which violates the heuristic **"Consistency and standards"**. For example, Ctrl + U in many text editor represents for a underlining function, while in Overleaf it is used for uppercase letters. This would decrease the usability of shortcuts in Overleaf and users need to spend more time to recover from errors.

4 Second Evaluation Phase

4.1 Evaluation Process

2 group meetings were held. Each member presented his or her individual table with all positive features and negative features found and gave corresponding evidence. For each feature or issue delivered, other team members were supposed to show agreement if they had successfully found the same point. Moreover, each participant gave top 5 features and top 10 issues listed in their own summary tables.
An aggregated table was constructed in a shared document. Firstly we wrote down our own top list so there were more than 5 features and more than 10 issues initially. Secondly we cut down to 5 and 10 respectively based on agreement results.
For a more comprehensive aggregated table, we add an additional step: other members can argue for their features which were not ranked on the top. Even though the initial agreement of some features was not enough to keep them in the top list, they could win a place in the top list if other people were convinced by justifications. *Those features are listed in the table with "Argued".*

4.2 Aggregated Top 5 Positive Features and Top 10 Issues Summary Table

Top List for identified Features and Issues Group Summary Table				
Order	Top 5 Positive Features	Agreement	Top 10 Issues	Agreement
1	Universally recognisable icons and buttons	4	Non-descriptive error messages	4
2	Helping tips are offered when the mouse is hovering over buttons	3	Placement of Menu and Back buttons leads to wrong clicking	3
3	Loading animation when opening or compiling a document *(Argued)*	1	No Undo and Redo buttons	2
4	Collaboration mode enables multiple authors to distinguish each other by cursors in different colours in real-time writing process	3	Very few actions and features on the system are well explained or even linked to documentation (For example, archive)	2
5	Use little jargon in helping documentation and instructions	1	Meaning of the Back button on the top left is ambiguous: no tool tip when hovered over by mouse and the icon is not fully intuitive	3
6			Users need to recall if the file is modified since last compiled. If the file is large, then a new compilation would be time-consuming	1

7			Cancel and Delete buttons are close to each other in the confirmation box invoked by ”delete”	1
8			Hot keys are not very externally consistently	4
9			Users cannot access the documentation directly from the project screen	2
10			Misleading leftward and rightward arrow buttons located between the source file and the PDF file: users often click them for full screen	1

5 Conclusion

Overall, although Overleaf involves many drawbacks, it is still a good tool to generate a formal paper. Those drawbacks and issues can be improved in the future: for ambiguous error messages, developers can add a link to documentation where potential solutions for this error are available; for the misleading ”Back” button, tips for mouse hovering and an intuitive icon can be added, and they can reverse the placement of ”Menu” and ”Back”; for missing Undo and Redo buttons, they can add those two buttons on the editing page; developers can add a ”*” on the ”Recompile” button if the newest document is modified, so users do not need to compile when there is no ”*” showing; developers can add buttons or hot keys for ”go to last cursor position” and ”auto formatting”.

References

[1] Nielsen, J. (1995). 10 Heuristics for User Interface Design. *[online] Nielsen Norman Group*, 10 Oct 2019, `https://www.nngroup.com/articles/ten-usability-heuristics/`.

A Appendix: Identified Features and Issues Individual Full Table

Identified Features and Issues **Individual Full Table**			
Heuristic	Identified Feature or Problem	Brief Justification	Agreement
1	Feature 1: Warning dialogue when losing connection to online platform	A white box will pop up to notify users about connection loss and an orange box will pop up showing ”reconnecting”. This shows the current status of Internet connection	4
	Feature 2: Multi-user node enables visibility of individual progress by different cursor positions in different colours	User A’s cursor can be yellow and User B’s cursor can be blue. The cursor will jump to the user A’s position after clicking User A’s profile on the top. It shows individual status.	1
	Feature 3: User can be notified when a new chatting message from other users is sent by a red dot on the top right corner	The visible red dot shows the current chatting status for new messages	1

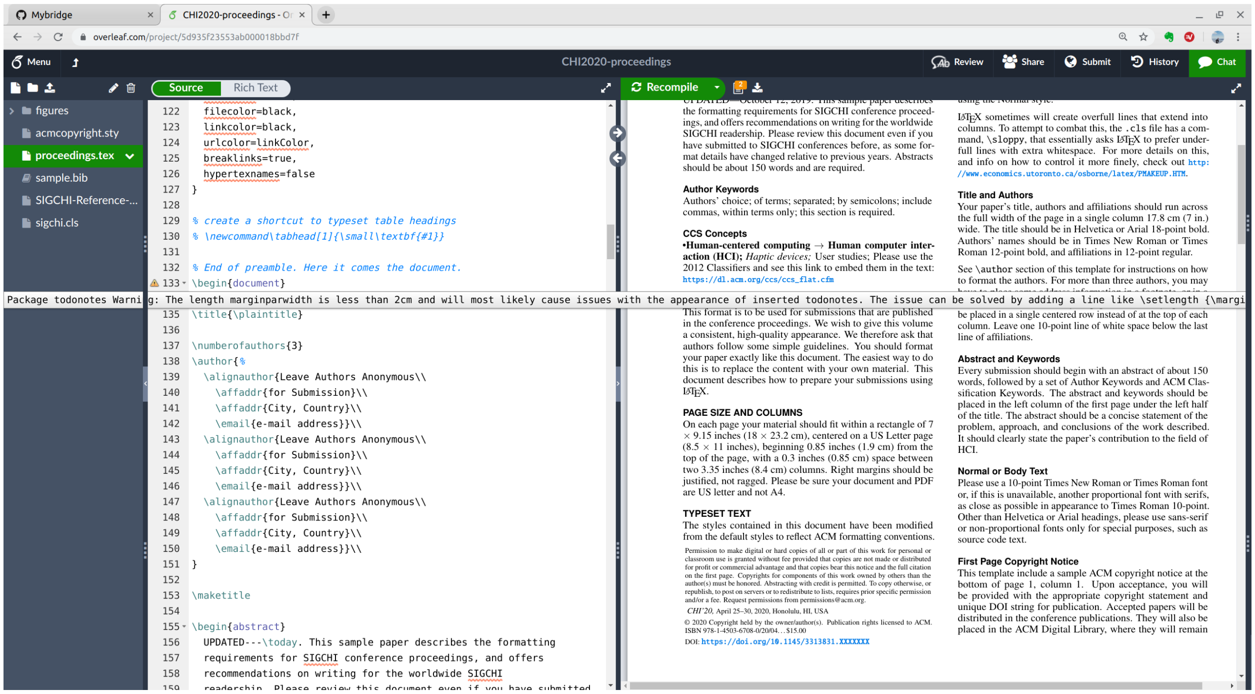
	Problem 1: Users easily forgets whether the compiled PDF is the newest version.	Nothing to inform users of whether the current file is compiled or modified, hence users may compile it many times even it is the newest version. Recompiling can be time-consuming for large files.	1
2	Feature 1: Universally recognisable icons and buttons	Most of them are shown by intuitive icons and tool tips can show up when the mouse is hovering over them; they are grouped and located together logically. Those buttons are more realistic and familiar to users	4
	Feature 2: Rich text mode offers a more organisable format	Rich text mode automatically formats the document and omits many type names, so the source document looks more elegant. This mode can make information in a natural way	1
	Problem 1: No automatic formatting hot key for source file	Contrast to the Feature 2, there is no button or shortcut can be used by users to format the source file, which easily leads to a messy format.	1
3	Feature 1: Undo and redo shortcuts are available for expert users to restore a previous state	Users can leave their unwanted states by clicking undo and redo, which endows users freedom and control over the system	4
	Problem 1: No undo and redo buttons for novice users	Novice users cannot leave their unwanted states by visible buttons after a typo typed	2
4	Feature 1: Options and buttons have identical or similar functions to buttons in other text editor techniques	In Microsoft Word and other editing tools, buttons, such as ”new file”, ”new folder”, exist. Users can be familiar with them quickly when switching between those different but consistent platforms	3
	Feature 2: Some shortcuts are same as shortcuts in other techniques.	In Microsoft Word and other editing tools, some shortcuts, such as ”Ctrl + C” for copying and ”Ctrl + V” for pasting, are the same as shortcuts in Overleaf. Thus, Overleaf is consistent externally.	1
	Feature 3: The online chatting interface looks similar and consistent with other chatting interfaces	Users can have their own color to present their messages and the layout of the whole chatting room is similar to Messenger (Facebook), Snapchat, etc. The chatting room is consistent externally	1

	Problem 1: The order of 2 buttons: Menu and Home buttons on the top left corner are confusing: other techniques usually put Home button on the the top left	In other techniques for editing document or other websites, they often put ”Home” or ”Back” button on the top-left most spot with an intuitive icon. However, in Overleaf the same spot holds ”Menu” for properties frame, which is misleading and many people made mistakes	3
	Problem 2: Misleading leftward and rightward arrow buttons located between the source file and the PDF file: users often click them for full screen	The rightward and leftward buttons give users a first impression that they can extend the current frame to the full-screen size. However, they merely present ”go to the PDF / code line”	1
5	Feature 1: Confirmation boxes before ”delete”	A proper confirmation box popped out after user clicks ”delete” tolerates an accidental clicking. Therefore, users can avoid severe consequences in the first time	3
	Feature 2: Spelling check with wrong words underlined by red lines	Spelling check prevents users from compiling / submitting with typos or grammar errors in the first time in a visible way	4
	Problem 1: A wrong type would not invoke an error triangle until compilation (For example, \textbfffff)	A correct type to bold texts is \textbf{}. If user types a \textbffffff{}, it would not give proper warnings. Only after compilation, users can find a red error triangle appearing in the side bar. This cannot prevent errors from occurring in the first time	1
	Problem 2: Logging out immediately without a confirmation dialog for users after users click ”Log out”	If users open ”Account” in the project home page and click ”Log out”, then they will be directed to logging out page without a confirmation box saying ”Are you sure you want to log out? ”. This would not tolerate an accidental clicking	2
6	Feature 1: Auto-completion and suggestion lists available	When users are typing ”\begin” then a drop-down list will appears immediately, with many suggested types for auto-fill. Users do not need to remember some long types precisely	1
	Feature 2: Necessary buttons are visible (such as download) to reduce memory load	In Microsoft Word, if users want to download the current document, they need to recall the position of download option hiding in the ”File” button. In Overleaf, a visible ”download” button is located on the top so users easily recognise it without memory load for it	1
	Feature 3: Log files storing the previous logs	If users want to know previous logs, they can see them in the ”log” dialogue rather than recalling them	2

	Feature 4: Double clicking a word in PDF leads to the corresponding position of the word in source file	Users do not need to remember or sequentially search for the word or line in the source file. This reduces their memory load	3
	Feature 5: History window for checking historical modifications and comparing different versions	Users can recognise previous modifications rather than remembering them	2
	Problem 1: Users easily forgets whether the compiled PDF is the newest version (see Heuristic 1: Problem 1)	Nothing to inform users of whether the current file is compiled or modified, hence users may compile it many times even it is the newest version. Recompiling can be time-consuming for large files.	1
	Problem 2: No shortcut to jump back to the last cursor position	Users sometimes need to backtrack to the last position they modified, while there is no shortcut provided by Overleaf can perform this function. Users have to recall the last cursor position	1
7	Feature 1: Some shortcuts are same as shortcuts in other techniques (see Heuristic 4: Feature 2)	In Microsoft Word and other editing tools, some shortcuts, such as ”Ctrl + C” for copying and ”Ctrl + V” for pasting, are the same as shortcuts in Overleaf. Thus, Overleaf is consistent externally	1
	Feature 2: Novice users can use the default properties while expert users can modify properties by clicking Menu button	It saves time for expert users but also it is flexible for novice users since they can use the default setting.	1
	Feature 3: Double clicking a word in PDF leads to the corresponding position of the word in source file (see Heuristic 6: Feature 4)	Users do not need to remember or sequentially search for the word or line in the source file. This reduces their memory load	3
	Problem 1: No shortcut to jump back to the last cursor position (see Heuristic 6: Problem 2)	Users sometimes need to backtrack to the last position they modified, while there is no shortcut provided by Overleaf can perform this function. Users have to recall the last cursor position	1
8	Feature 1: Only relevant buttons appearing	Most buttons in the main page perform important roles in Overleaf, which shows a minimalist design	3
	Feature 2: Rich text mode offers a more organisable format (see Heuristic 2: Feature 2)	Rich text mode automatically formats the document and omits many type names, so the source document looks more elegant. This mode can make information in a natural way	1
	Problem 1: Duplicate buttons for full screen: the full screen button and the ”hide PDF” button located between the source file and the PDF file	They both lead to a full-screen source file layout, but it violates the ”minimalist”	1

	Problem 2: No automatic formatting hot key for source file (see Heuristic 2: Problem 1)	There is no button or shortcut can be used by users to format the source file, which easily leads to a messy format rather than an aesthetic one.	1
9	Feature 1: Warning dialogue when losing connection to online platform (see Heuristic 1 Feature 1)	A white box will pop up to notify users about connection loss and an orange box will pop up showing "reconnecting". Users can recognise and diagnose the occurrence of errors by checking their own Internet connection	4
	Feature 2: From history window users can recover	Uses can inspect a previous version and then "download as the current version", which enables users to recover from errors if their want to go back to a previous state	2
	Feature 3: Logs and output files button can bring a list of error and warning messages with potential solutions to them	It helps users to recognise and diagnose errors, and if users click on one error message, the cursor will jump to the specific problematic line in source file	2
	Feature 4: Red error triangles and yellow warning triangles appearing in the left side bar	It can notify users when errors occur in a visible way. When the mouse is hovering over those triangles, error messages show up to help users diagnose errors	4
	Problem 1: Some error messages are confusing and ambiguous	Messages as "Undefined Control Sequence" do not offer useful information and do not offer a direct link to documentation for explanation. Users need to inspect every case which may cause this error	4
	Problem 2: Some error messages overflow the current window so users only see truncated messages (see Figure 1)	Shown by a screen shot appended in to appendix	2
10	Feature 1: Adequate helping documentation	Documentation in Overleaf is relatively comprehensive and complete, including solutions to common errors and some brief introduction to types, sometimes with videos explained.	4
	Feature 2: Tool tips when the mouse is hovering over each button	Those small tips occur by mouse hovering help users quickly understand functions of buttons	3
	Problem 1: Some error messages are confusing and ambiguous see(Heuristic 9: Problem 1)	Messages as "Undefined Control Sequence" do not offer useful information and do not offer a direct link to documentation for explanation. Users need to inspect every case which may cause this error	4

B Appendix: Image Evidence for Heuristic 9 Problem 2



```
  filecolor=black,
  linkcolor=black,
  urlcolor=linkColor,
  breaklinks=true,
  hypertexnames=false
}

% create a shortcut to typeset table headings
% \newcommand\tabhead[1]{\small\textbf{#1}}

% End of preamble. Here it comes the document.
\begin{document}
```

```
Package todonotes Warning: The length marginparwidth is less than 2cm and will most likely cause issues with the appearance of inserted todonotes. The issue can be solved by adding a line like \setlength {\margi
```

```
\title{\plaintitle}

\numberofauthors{3}
\author{%
  \alignauthor{Leave Authors Anonymous\\
    \affaddr{for Submission}\\
    \affaddr{City, Country}\\
    \email{e-mail address}}\\
  \alignauthor{Leave Authors Anonymous\\
    \affaddr{for Submission}\\
    \affaddr{City, Country}\\
    \email{e-mail address}}\\
  \alignauthor{Leave Authors Anonymous\\
    \affaddr{for Submission}\\
    \affaddr{City, Country}\\
    \email{e-mail address}}\\
}

\maketitle

\begin{abstract}
  UPDATED---\today. This sample paper describes the formatting
  requirements for SIGCHI conference proceedings, and offers
  recommendations on writing for the worldwide SIGCHI
  readership. Please review this document even if you have submitted
```

UPDATED—October 12, 2019. This sample paper describes the formatting requirements for SIGCHI conference proceedings, and offers recommendations on writing for the worldwide SIGCHI readership. Please review this document even if you have submitted to SIGCHI conferences before, as some format details have changed relative to previous years. Abstracts should be about 150 words and are required.

Author Keywords
Authors' choice; of terms; separated; by semicolons; include commas, within terms only; this section is required.

CCS Concepts
•**Human-centered computing** → **Human computer interaction (HCI);** *Haptic devices;* User studies; Please use the 2012 Classifiers and see this link to embed them in the text: https://dl.acm.org/ccs/ccs_flat.cfm

This format is to be used for submissions that are published in the conference proceedings. We wish to give this volume a consistent, high-quality appearance. We therefore ask that authors follow some simple guidelines. You should format your paper exactly like this document. The easiest way to do this is to replace the content with your own material. This document describes how to prepare your submissions using LaTeX.

PAGE SIZE AND COLUMNS
On each page your material should fit within a rectangle of 7 × 9.15 inches (18 × 23.2 cm), centered on a US Letter page (8.5 × 11 inches), beginning 0.85 inches (1.9 cm) from the top of the page, with a 0.3 inches (0.85 cm) space between two 3.35 inches (8.4 cm) columns. Right margins should be justified, not ragged. Please be sure your document and PDF are US letter and not A4.

TYPESET TEXT
The styles contained in this document have been modified from the default styles to reflect ACM formatting conventions.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
CHI'20, April 25–30, 2020, Honolulu, HI, USA
© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ISBN 978-1-4503-6708-0/20/04...$15.00
DOI: https://doi.org/10.1145/3313831.XXXXXXX

using the Normal style.

LaTeX sometimes will create overfull lines that extend into columns. To attempt to combat this, the .cls file has a command, \sloppy, that essentially asks LaTeX to prefer underfull lines with extra whitespace. For more details on this, and info on how to control it more finely, check out http://www.economics.utoronto.ca/osborne/latex/PMAKEUP.HTM.

Title and Authors
Your paper's title, authors and affiliations should run across the full width of the page in a single column 17.8 cm (7 in.) wide. The title should be in Helvetica or Arial 18-point bold. Authors' names should be in Times New Roman or Times Roman 12-point bold, and affiliations in 12-point regular.

See \author section of this template for instructions on how to format the authors. For more than three authors, you may have to place some address information in a footnote, or in a be placed in a single centered row instead of at the top of each column. Leave one 10-point line of white space below the last line of affiliations.

Abstract and Keywords
Every submission should begin with an abstract of about 150 words, followed by a set of Author Keywords and ACM Classification Keywords. The abstract and keywords should be placed in the left column of the first page under the left half of the title. The abstract should be a concise statement of the problem, approach, and conclusions of the work described. It should clearly state the paper's contribution to the field of HCI.

Normal or Body Text
Please use a 10-point Times New Roman or Times Roman font or, if this is unavailable, another proportional font with serifs, as close as possible in appearance to Times Roman 10-point. Other than Helvetica or Arial headings, please use sans-serif or non-proportional fonts only for special purposes, such as source code text.

First Page Copyright Notice
This template include a sample ACM copyright notice at the bottom of page 1, column 1. Upon acceptance, you will be provided with the appropriate copyright statement and unique DOI string for publication. Accepted papers will be distributed in the conference publications. They will also be placed in the ACM Digital Library, where they will remain

Figure 1: Heuristic 9 Problem 2: Error message overflows the current window