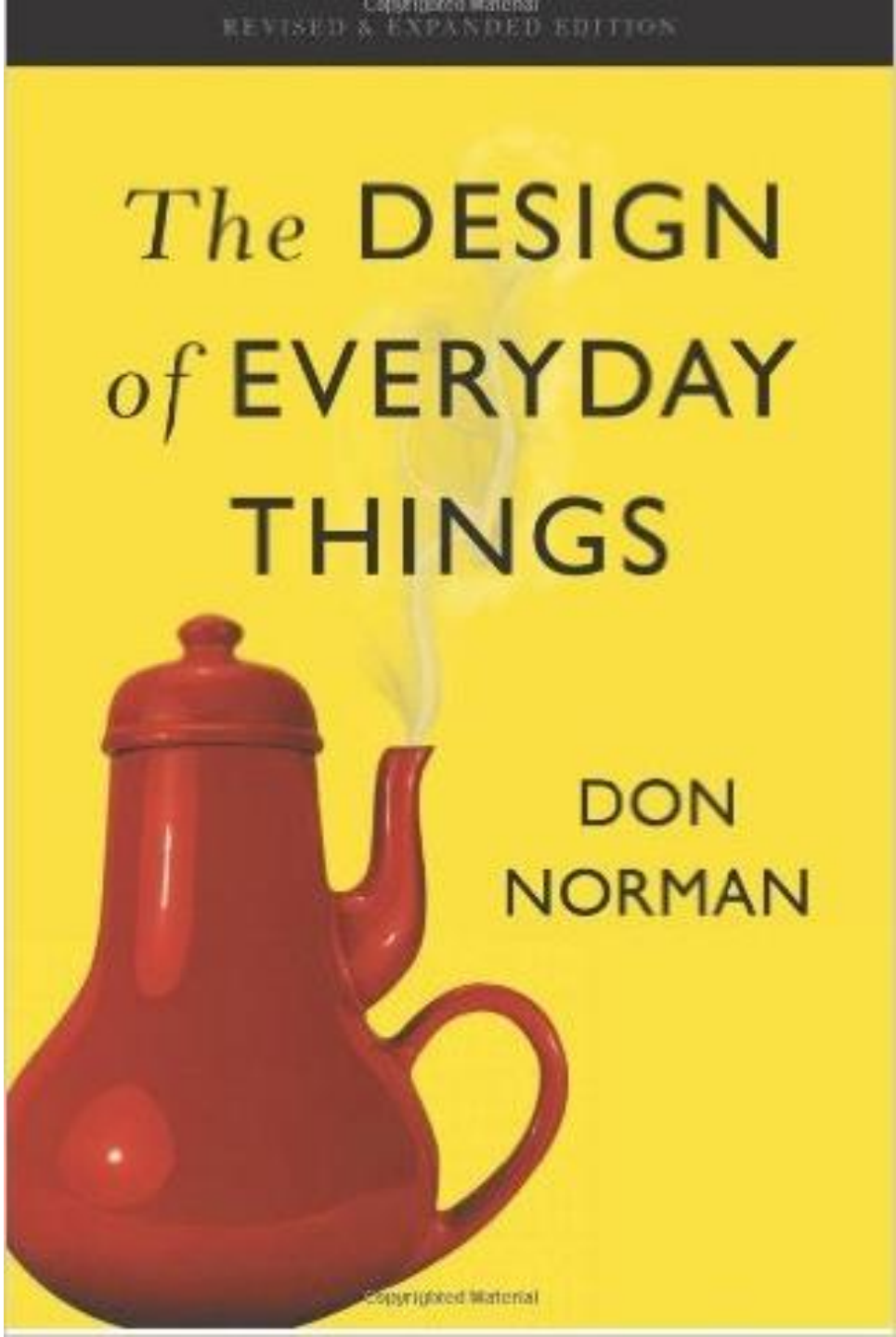
HCI: HEURISTICS AND COGNITIVE WALKTHROUGH

Dr Kami Vaniea

First, the news…



Tutorials

Coursework 1 Q&A

- Think aloud normally recommends 5 evaluations why does the coursework call for only 1?
  - This is a 10 credit course I am trying to not over-work you.
  - The primary point of marking on the coursework is if you can accurately conduct the usability method. More subjects will produce more data, but you only need the number listed to demonstrate that you can do the method

Usability Inspections

Inspections

- Several kinds.
- Experts use their knowledge of users & technology to review software usability.
- Expert critiques can be formal or informal.
- Heuristic evaluation is a review guided by a set of heuristics.
- Walkthroughs involve stepping through a pre-planned scenario noting potential problems.

Heuristic Evaluation

- Basic idea: Have one or more experts evaluate an interface based on a common set of criteria
- Heuristic Evaluation is very easy to do
- Pros
  - Can be done by even a single person
  - No ethics, recording, or other human-related problems
  - Minimal expense to find a large number of potentially expensive problems
- Cons
  - Experts are not the same as end users, they will miss some things
  - Heuristics are the most common types of problems but they do not represent all problems

3 stages for doing heuristic evaluation

- Briefing session to tell experts what to do.
- Evaluation period of 1-2 hours in which:
  - Each expert works separately;
  - Take one pass to get a feel for the product;
  - Take a second pass to focus on specific features.
- Debriefing session in which experts work together to prioritize problems.

No. of evaluators & problems

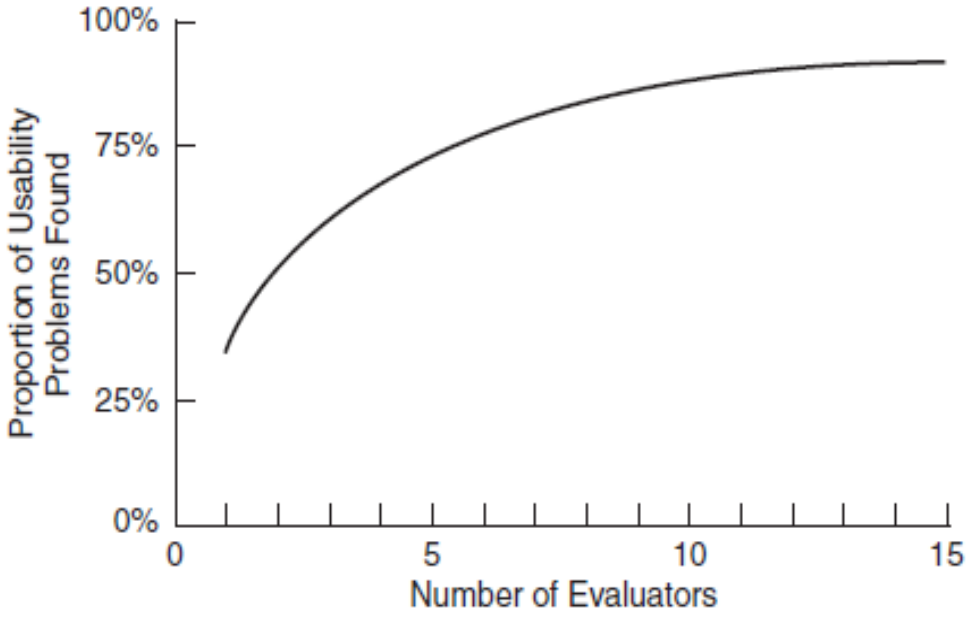


Figure 15.1 Curve showing the proportion of usability problems in an interface found by heuristic evaluation using various numbers of evaluators. The curve represents the average of six case studies of heuristic evaluation

Source: Usability Inspection Methods, J. Nielson & R.L. Mack ©1994. Reproduced with permission of John Wiley & Sons Inc.

Number of evaluators

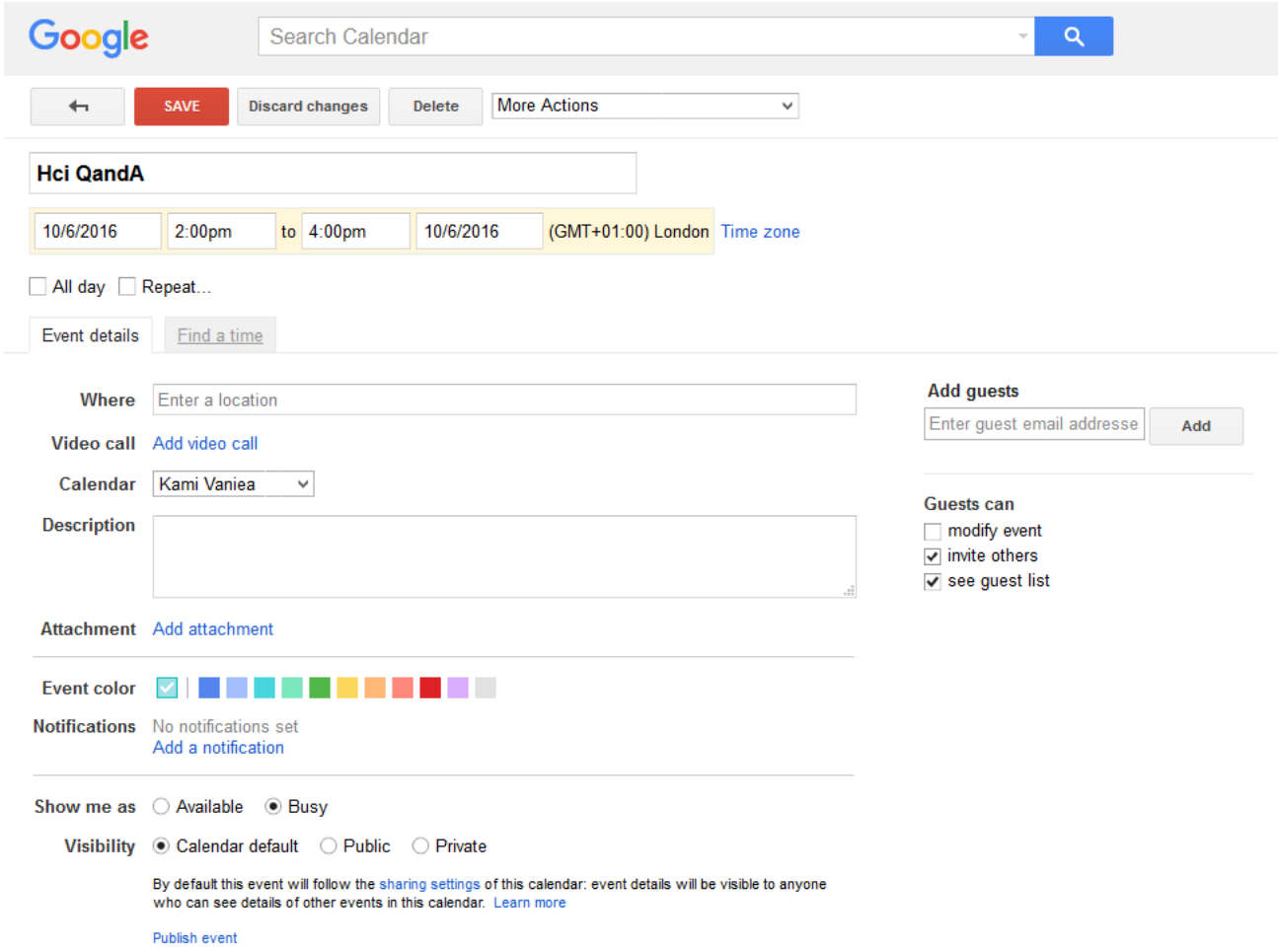
- Nielsen suggests that on average 5 evaluators identify 75-80% of usability problems.
- Cockton and Woolrych (2001) point out that the number of users needed to find 75-80% of usability problems depends on the context and nature of the problems.

Neilson's 10 Heuristics

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility an deficiency of use
8. Aesthetics and minimalist design
9. Help users recognize, diagnose, and recover from errors
10. Help and documentation

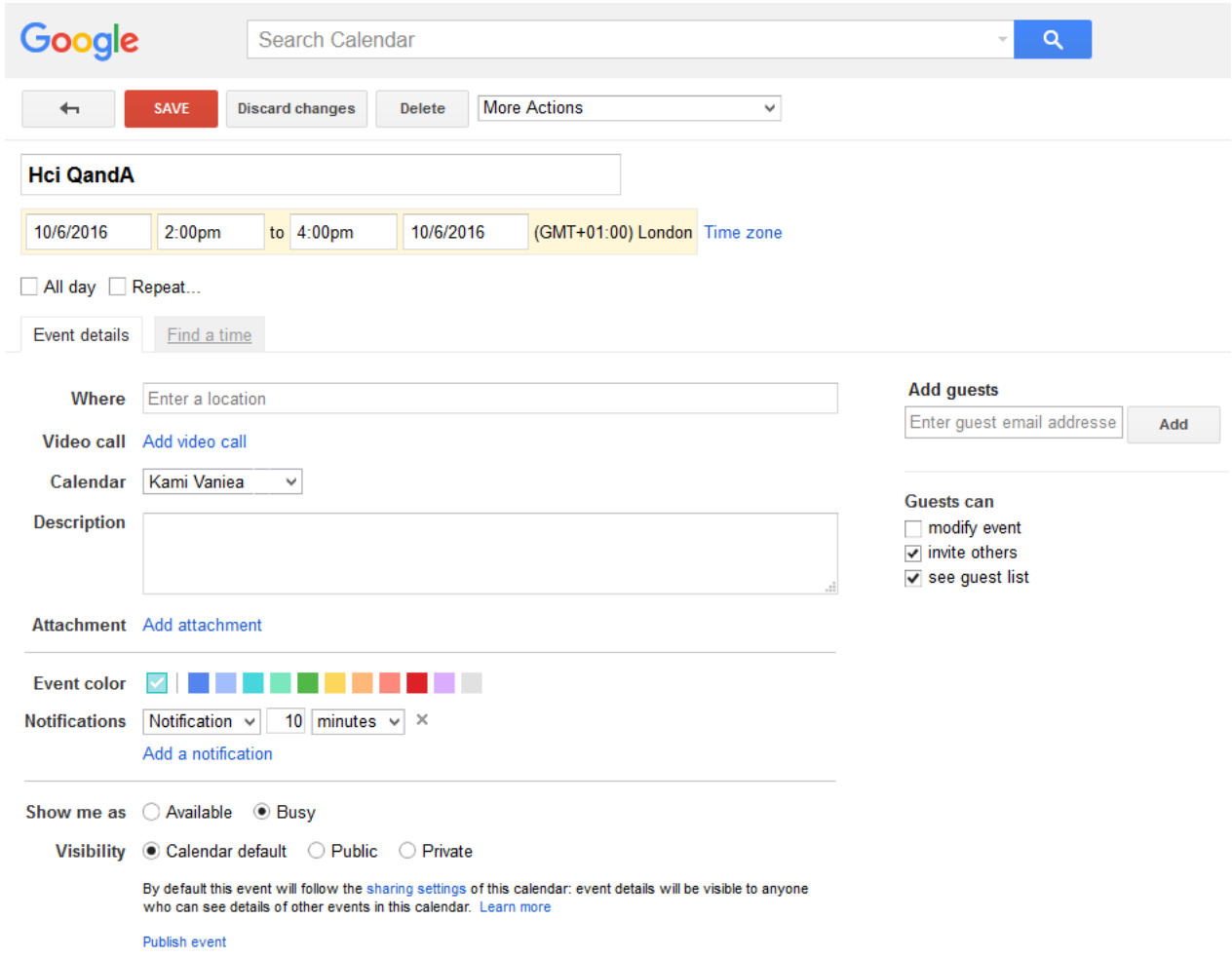
Visibility of system status

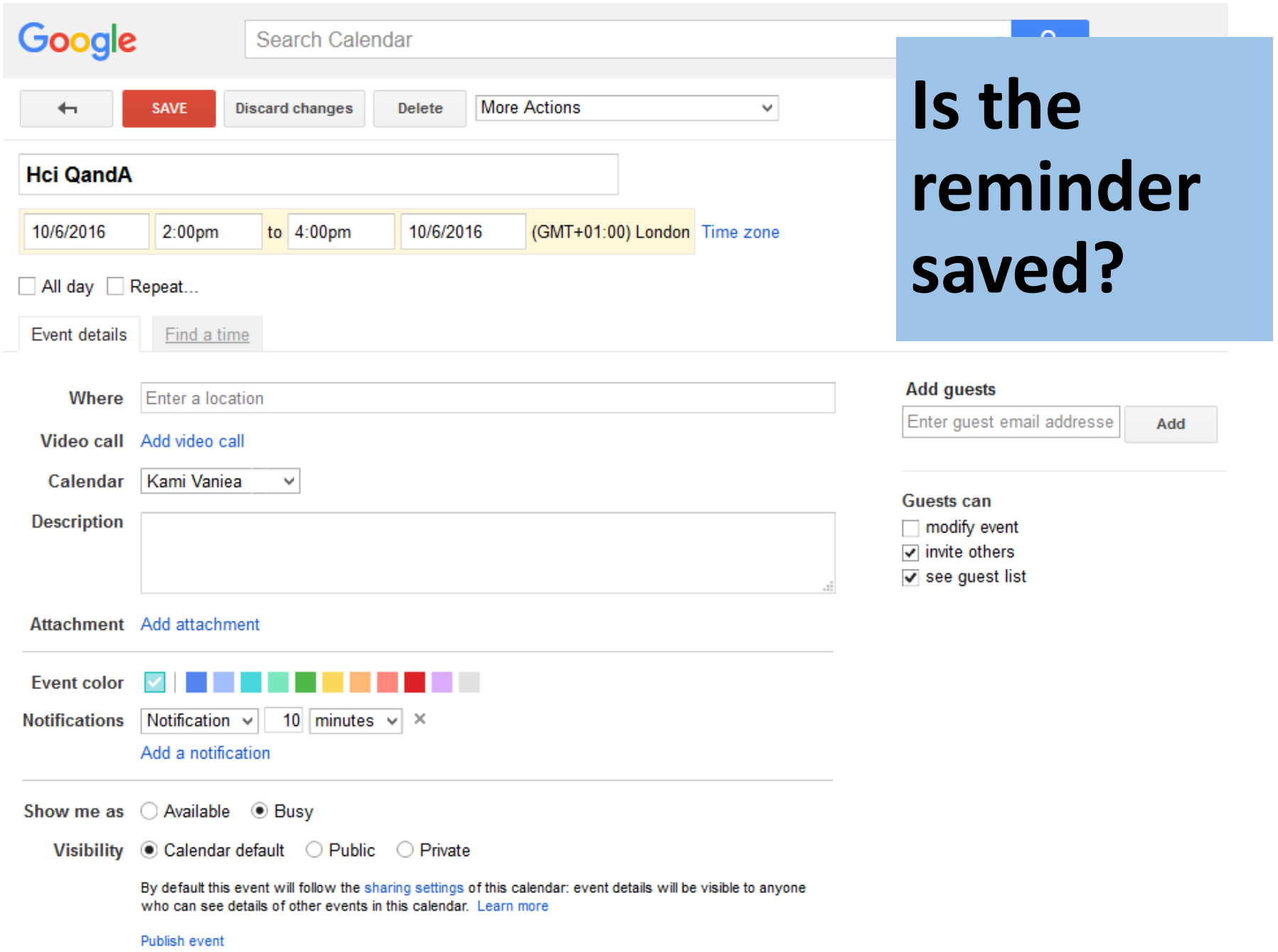
Me adding the Q&A session to my Google calendar



Visibility of system status

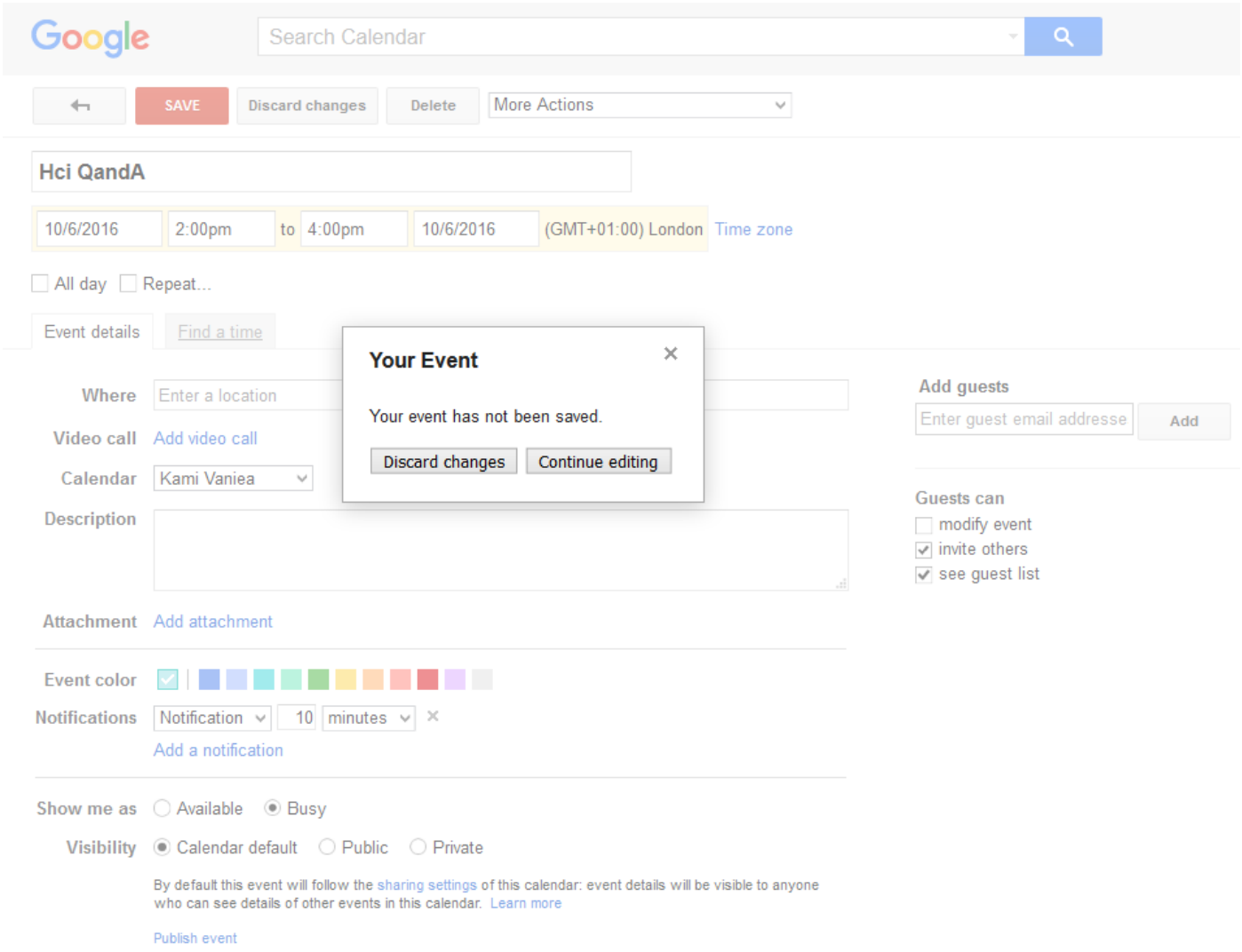
Better add a reminder or I might forget to go




Google
Search Calendar
SAVE
Discard changes
Delete
More Actions
Hci QandA
10/6/2016
2:00pm
to
4:00pm
10/6/2016
(GMT+01:00) London
Time zone
All day
Repeat...
Event details
Find a time
Where
Enter a location
Video call
Add video call
Calendar
Kami Vaniea
Description
Attachment
Add attachment
Event color
Notifications
Notification
10
minutes
Add a notification
Show me as
Available
Busy
Visibility
Calendar default
Public
Private
By default this event will follow the sharing settings of this calendar: event details will be visible to anyone who can see details of other events in this calendar. Learn more
Publish event
Add guests
Enter guest email addresse
Add
Guests can
modify event
invite others
see guest list
Is the reminder saved?

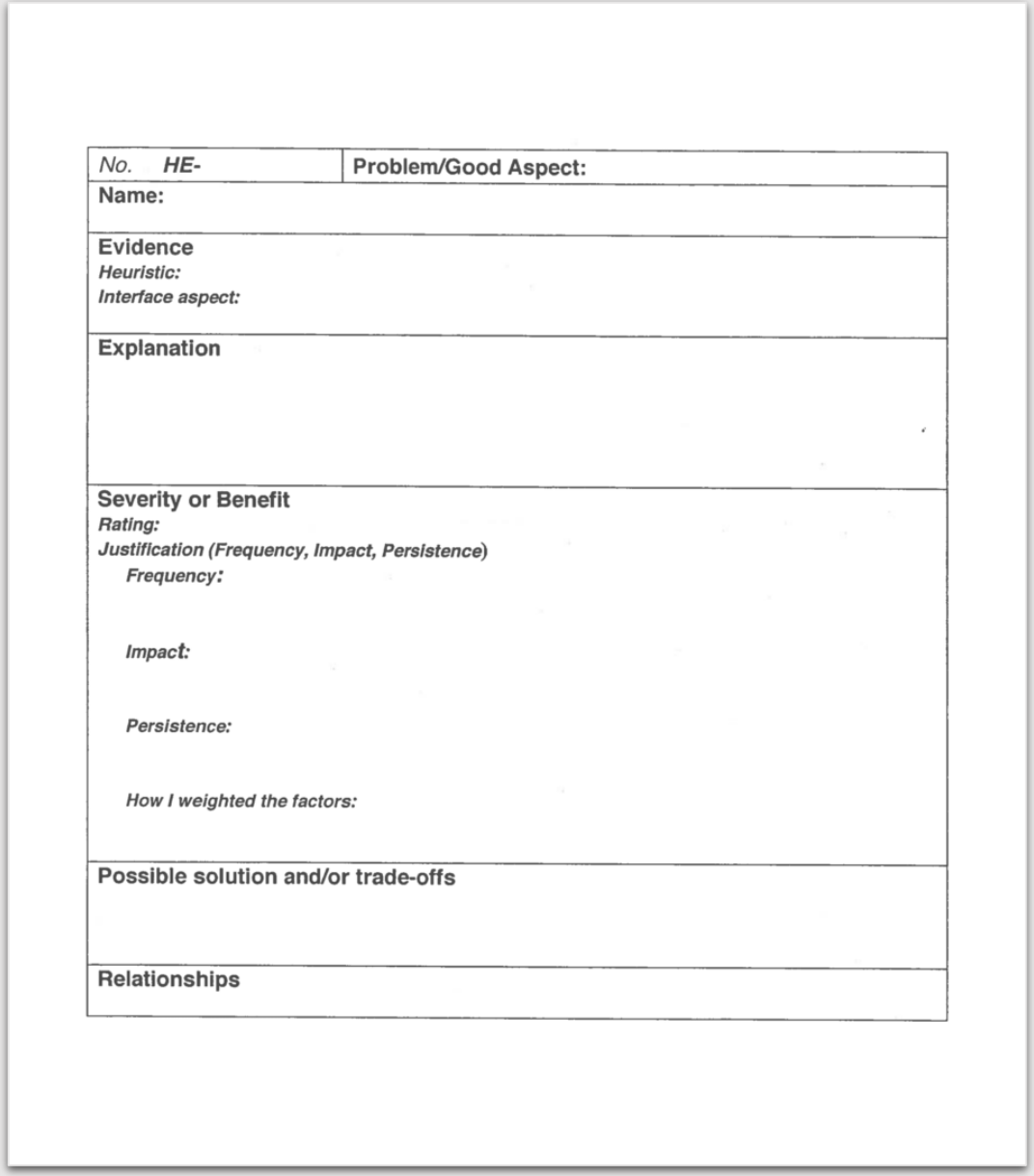
Visibility of system status

I click the back button without clicking “save” and get a warning



Usability Aspect Reports (UAR)

- Similar to a bug report, but for usability issues
- Can be about good or bad features
- Should link to a heuristic



No. ***HE-***	**Problem/Good Aspect:**
Name:	
Evidence ***Heuristic:*** ***Interface aspect:***	
Explanation	
Severity or Benefit ***Rating:*** ***Justification (Frequency, Impact, Persistence)*** ***Frequency:*** ***Impact:*** ***Persistence:*** ***How I weighted the factors:***	
Possible solution and/or trade-offs	
Relationships	

HE-01	**Problem/Good Aspect:** Problem

Name: Saved status not visible for calendar changes

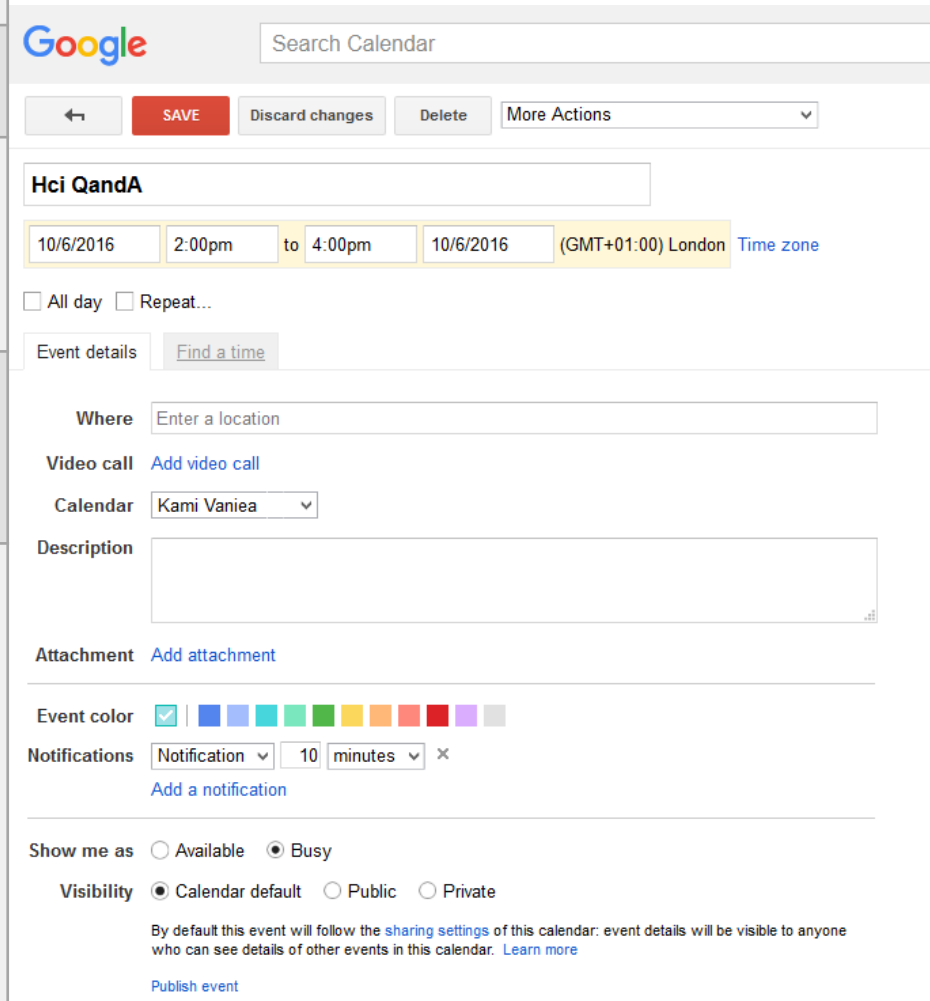
Evidence
Heuristic: Visibility of status
Interface aspect:

Explanation: When a calendar event element is changed it is not clear if it is automatically saved or not. As a result a user may try and leave the page when it is not saved.

Severity or Benefit
Rating: Low
Justification: A warning box pops up preventing accidental loss of data
Frequency: Medium
Impact: Low
Persistence: High (happens every time)
How I weight the factors: The error is very recoverable and the warning is clear, so this may be an issue but it is a low importance one.

Possible solution and/or tradeoff: Automatic saving is possible, but that may lead to other issues

Relationships: None



Neilson's 10 Heuristics

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility an deficiency of use
8. Aesthetics and minimalist design
9. Help users recognize, diagnose, and recover from errors
10. Help and documentation

Match between system and the real world

- The interface should use concepts, language and real-world conventions that are familiar to the user.
- Why
  - The user already has knowledge from the outside world. A user interface can leverage that knowledge
  - If the interface does not match the way the world typically works people will become confused

Next Back

User control and freedom

- Allow the user to have control of the interaction. Users should be able to undo actions, exit from any sequence of actions, and not be forced into a series of actions.
- Why
  - Users make errors sometimes
  - They need the ability to go back and correct the errors

Consistency and standards

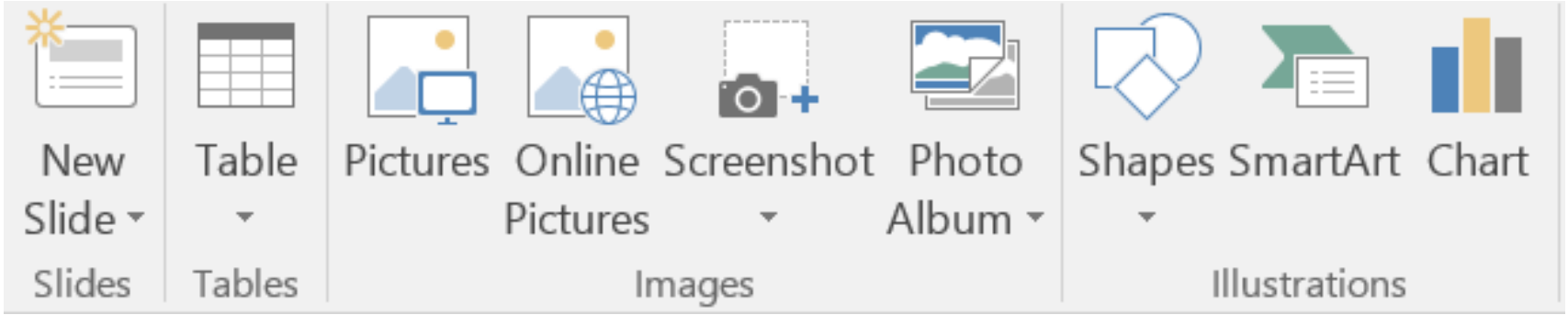
- Information that is the same should appear to be the same
- Information that is different should be expressed differently
- Developers need to know the conventions being used in the software
- Why
  - Similar to the real world heuristic, people can leverage things they already know
  - They will expect that something they learned will continue to be true

Error prevention

- If possible, prevent errors from happening in the first place
- Similar to visibility of system status, but specifically involves preventing an error from happening
- Example: if the user needs to select 3 things, don't wait till the next screen to tell them that they have selected 4
- Why
  - Users are not machines, they do not always perceive all the information available and they can temporarily forget things
  - Computers are really good at using all the information available and remembering the last few things

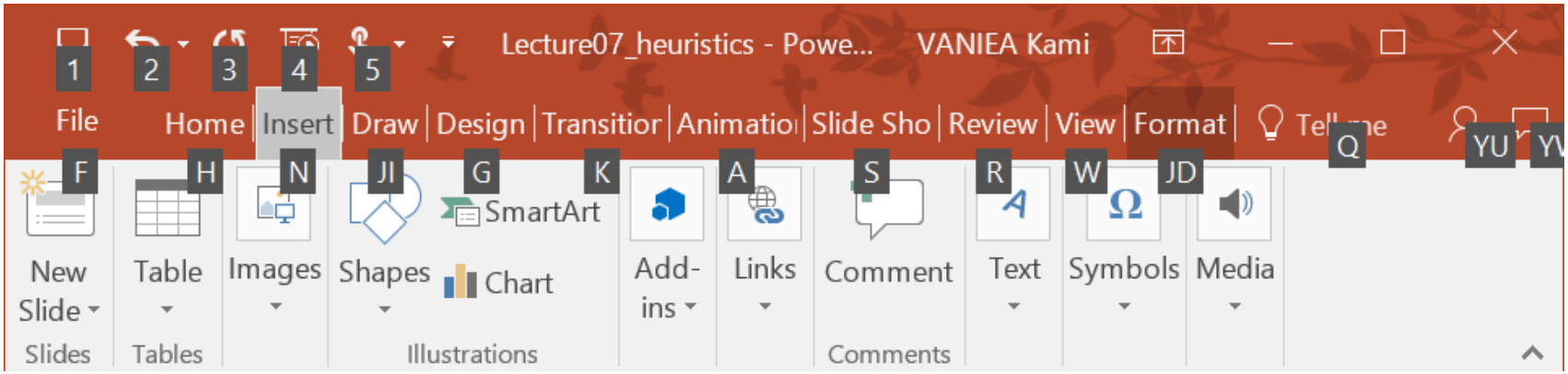
Recognition rather than recall

- Show all the options available to the user rather than expecting them to remember them all
- Do not require users to remember information from one screen to the next
- Why
  - People are less good at remembering (recall) than they are at recognizing (recognition)



Flexibility and efficiency of use

- Experts should have a way to use the interface faster or more efficiently
- Design should have accelerators like keyboard shortcuts to allow skilled users to move faster
- Why
  - Using the mouse is MUCH slower than the keyboard. Users who know what they want should be able to find it quickly and efficiently



Aesthetics and minimalist design

- Get rid of clutter
- See most of the “Don’t make me think” book
- Why
  - The more things there are to look at, the harder it is for a user to process the data

Help users recognize, diagnose, and recover from errors

- Error messages should be clear, written in plain English, explain the problem, give constructive advice on how to solve the problem
- Why
  - Errors should only be shown to users when the system can no longer make a choice on their behalf. The error needs to be clear about what it is the user needs to do or provide input on

Help and documentation

- Unless the system is extremely simple, some people will need help documentation
- Why
  - People learn about things in different ways. Some people learn by playing around and pushing buttons, other people learn by reading. The system needs to support all people.

Heuristics for websites focus on key criteria (Budd, 2007)

- Clarity
- Minimize unnecessary complexity & cognitive load
- Provide users with context
- Promote positive & pleasurable user experience

Advantages and problems

- Few ethical & practical issues to consider because users not involved.
- Can be difficult & expensive to find experts.
- Best experts have knowledge of application domain & users.
- Biggest problems:
  - Important problems may get missed;
  - Many trivial problems are often identified;
  - Experts have biases.

Questions?