

HCI Support Card:

Creating and Using a Support Card for Education in
Human-Computer Interaction



PUC Minas



Lesandro Ponciano

@lesandrop

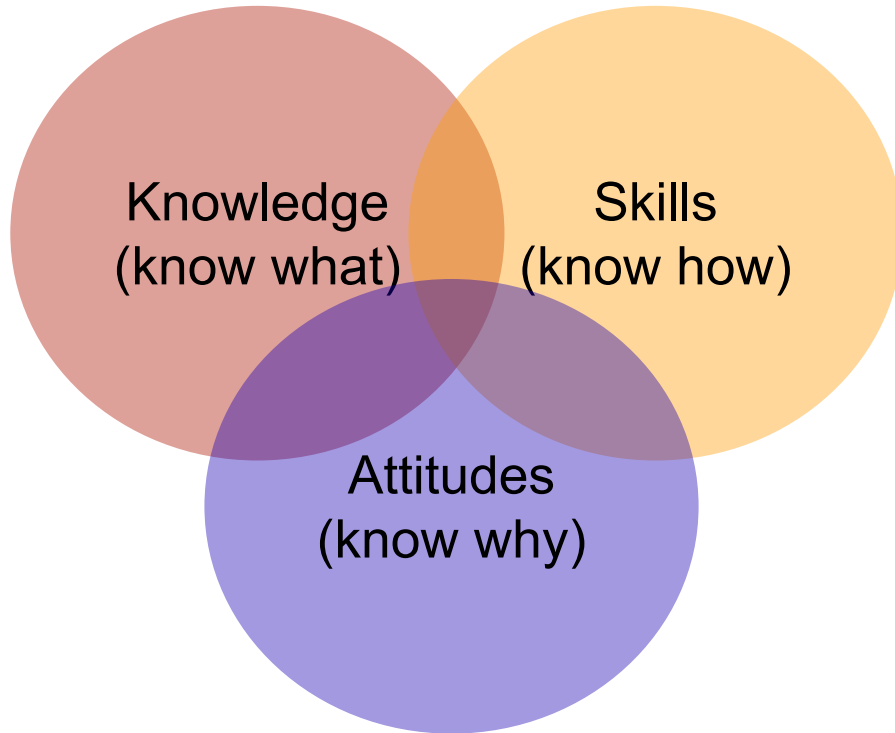
lesandrop@pucminas.br

HCI area:

- conceptual and theoretical frameworks
- guidelines, heuristics, rules, recommendations for design
- evaluation methods

Memorising information may not be necessary, especially if it is readily available for consultation

Knowledge, Skills and Attitudes in HCI Education



Lecture-based classes

Case history reviews

Problem solving cases

Decision-making cases

Support Cards:

- Guides that summarise a set of core information about a subject and are a quick way of accessing information about a subject or task
- Usually associated with the following terms: “quick reference card”, “pocket card”, “cheat sheet”

TABELA PERIÓDICA DOS ELEMENTOS

Tabuada da Multiplicação

0 x 0 = 0	1 x 0 = 0	2 x 0 = 0	3 x 0 = 0	4 x 0 = 0
0 x 1 = 0	1 x 1 = 1	2 x 1 = 2	3 x 1 = 3	4 x 1 = 4
0 x 2 = 0	1 x 2 = 2	2 x 2 = 4	3 x 2 = 6	4 x 2 = 8
0 x 3 = 0	1 x 3 = 3	2 x 3 = 6	3 x 3 = 9	4 x 3 = 12
0 x 4 = 0	1 x 4 = 4	2 x 4 = 8	3 x 4 = 12	4 x 4 = 16
0 x 5 = 0	1 x 5 = 5	2 x 5 = 10	3 x 5 = 15	4 x 5 = 20
0 x 6 = 0	1 x 6 = 6	2 x 6 = 12	3 x 6 = 18	4 x 6 = 24
0 x 7 = 0	1 x 7 = 7	2 x 7 = 14	3 x 7 = 21	4 x 7 = 28
0 x 8 = 0	1 x 8 = 8	2 x 8 = 16	3 x 8 = 24	4 x 8 = 32
0 x 9 = 0	1 x 9 = 9	2 x 9 = 18	3 x 9 = 27	4 x 9 = 36

5 x 0 = 0	6 x 0 = 0	7 x 0 = 0	8 x 0 = 0	9 x 0 = 0
5 x 1 = 5	6 x 1 = 6	7 x 1 = 7	8 x 1 = 8	9 x 1 = 9
5 x 2 = 10	6 x 2 = 12	7 x 2 = 14	8 x 2 = 16	9 x 2 = 18
5 x 3 = 15	6 x 3 = 18	7 x 3 = 21	8 x 3 = 24	9 x 3 = 27
5 x 4 = 20	6 x 4 = 24	7 x 4 = 28	8 x 4 = 32	9 x 4 = 36
5 x 5 = 25	6 x 5 = 30	7 x 5 = 35	8 x 5 = 40	9 x 5 = 45
5 x 6 = 30	6 x 6 = 36	7 x 6 = 42	8 x 6 = 48	9 x 6 = 54
5 x 7 = 35	6 x 7 = 42	7 x 7 = 49	8 x 7 = 56	9 x 7 = 63
5 x 8 = 40	6 x 8 = 48	7 x 8 = 56	8 x 8 = 64	9 x 8 = 72
5 x 9 = 45	6 x 9 = 54	7 x 9 = 63	8 x 9 = 72	9 x 9 = 81

SCRUM CHEAT SHEET

PROCESS

Scrum projects are organized in short iterations, during which the team works to add incremental value to the end product.

TASK SIZE

One task should fit within a single Sprint. Usually split into smaller tasks if exceeds

12 hrs

TASK ASSIGNMENT

Team self-assigns tasks by pulling them from the Sprint Backlog.

NEW TASKS IN ITERATION

Once the Sprint started, no new tasks can be added.

MEETINGS

Sprint Planning - a session for team and PO to determine what will be done in the next sprint.

Daily Scrum - a 15 minute standup where team members present what they have done and will do.

Sprint Review - team presentation of the Sprint results to the PO, checking if set goals were met.

Sprint Retrospective - discussion between team and SM about how the next Sprint can be improved.

1 PRIORITIZATION

Product Backlog is used to prioritize User Stories and to give clear direction for the team.

ROLES

Product Owner - represents the customer's interests and prioritizes requirements in backlog.

Scrum Master - facilitates Scrum application and communication between the team and the PO.

Team - The Team - Cross-functional group of people that self organizes to meet the Sprint goals.

Core responsibility: Delivering value

ITERATIONS

Scrum iterations are called Sprints. Planned and run one at a time, they usually last **1-4 weeks**

ESTIMATION

Must be done before every Sprint, either in hours or Story Points.

SCOPE LIMITS

The amount of work the team is going to do is limited by Sprints.

BOARD



TERMS

Product Backlog
First column on the board that holds all the prioritized user stories and represents a project plan curated by the Product Owner.

Sprint Backlog
A place for the team to divide the selected user stories into clear tasks and prepare for the Sprint.

Story Point
A measuring unit used instead of hours to estimate task duration. One story point can stand for any period of time or task size.

User Story
A short description of a feature that a customer is asking for. Usually written in the form: As a - I want ... So that ...

Burndown Chart
Shows how the number of uncompleted tasks is getting lower with time and can indicate issues within the process.

Team Velocity
Defines how much product backlog effort a team can handle in one sprint.

Usually combined out of 4 sections:

1. Product Backlog for prioritized user stories.
2. Sprint Backlog for planned tasks
3. Doing for tasks in progress
4. Done for completed tasks

Python 3 cheatsheet (the basics)



Interact with the user (input and output)

```
Print a message
print('Hello, world!')

Print multiple values (of different types)
ndays = 365
print('There are', ndays, 'in a year!')

Asking the user for a string
name = input('What is your name? ')

Asking the user for a whole number (an integer)
num = int(input('Enter a number: '))
```

Decide between options

```
Decide to run a block (or not)
x = 3
if x == 3:
    print('x is 3')

Decide between two blocks
mark = 80
if mark >= 50:
    print('pass')
else:
    print('fail')

Decide between many blocks
mark = 80
if mark >= 65:
    print('credit')
elif mark >= 50:
    print('pass')
else:
    print('fail')

*elif can be used without else
*elif can be used many times

Are two values equal?
x == 3
△ two equals signs, not one

Are two values not equal?
x != 3
Less than another?
x < 3
Greater than another?
x > 3
Less than or equal to?
x <= 3
Greater than or equal to?
x >= 3

The answer is a Boolean:
True or False
```

String manipulation

```
Convert to uppercase
msg = 'hello'
msg.upper()

Count and title
msg = 'hello'
print('howdy')

Also lower and title

Count a character in a string
msg = 'hello'
msg.count('l')

Replace a character or string
msg = 'hello'
msg.replace('l', 'x')

Delete a character or string
msg = 'hello'
msg.replace('l', '')

Is a character in a string?
'e' in msg

Is the string all lowercase?
msg.islower()

Is a string in another string?
'ell' in msg
```



Learn more in Intro, to Programming @ groklearning.com

Text (strings)

```
Single quoted
'perfect'

Double quoted
"credit"

Multi-line
"""Hello,
World!"""

Add (concatenate) strings:
'Hello' + 'World'

Multiply string by integer
len('Hello')

Convert string to integer
int('365')
```

Repeat a block (a fixed number of times)

```
Repeat a block 10 times
for i in range(10):
    print(i)

Sum the numbers 0 to 9
total = 0
for i in range(10):
    total = total + i
print(total)

Repeat a block over a string
for c in 'Hello':
    print(c)

Keep printing on one line
for c in 'Hello':
    print(c, end=' ')
print('!')

Repeat a block over list (or string) indices
msg = 'I grok Python!'
for i in range(len(msg)):
    print(i, msg[i])
```

Putting it together: Celsius to Fahrenheit converter

```
Ask the user for a temperature in degrees Celsius
celsius = int(input('Temp. in Celsius: '))

Calculate the conversion
fahrenheit = celsius*9/5 + 32

Output the result
print('Fahrenheit, 'Fahrenheit')
```

Variables

```
Creating a variable
celsius = 25

Using a variable
celsius*9/5 + 32
```

Whole numbers (integers)

```
Addition and subtraction
365 + 1 - 2

Multiplication and division
25*9/5 + 32

Powers (2 to the power of 8)
2**8

Convert integer to string
str(365)
```

Multidisciplinary ● Academic ● Professional



Research questions

1. how to draw a support card to be used as a didactic resource in HCI classroom?
2. what utilities do students perceive in using the card?
3. how useful is the card beyond the HCI classroom?

Process for Creating and Using the Support Card

To define the main requirements of the support card

Syllabus, interdisciplinary, useful to students in the classroom and homework, e **easy to handle and read**

To define the topics to be covered on the card

Topics that are **harder to remember**, and topics that serve as **anchors** for other topics

To organise the topics on the card

Organisation of the topics on the card must be **intuitive for students** so that they can use the card effectively

To use the card in the classroom

Hand-on activities, design, evaluation and creative tasks

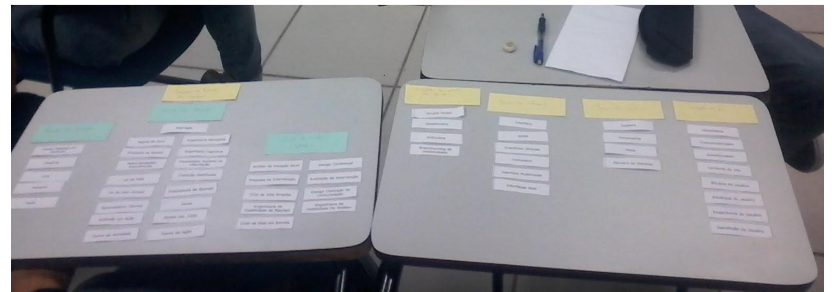
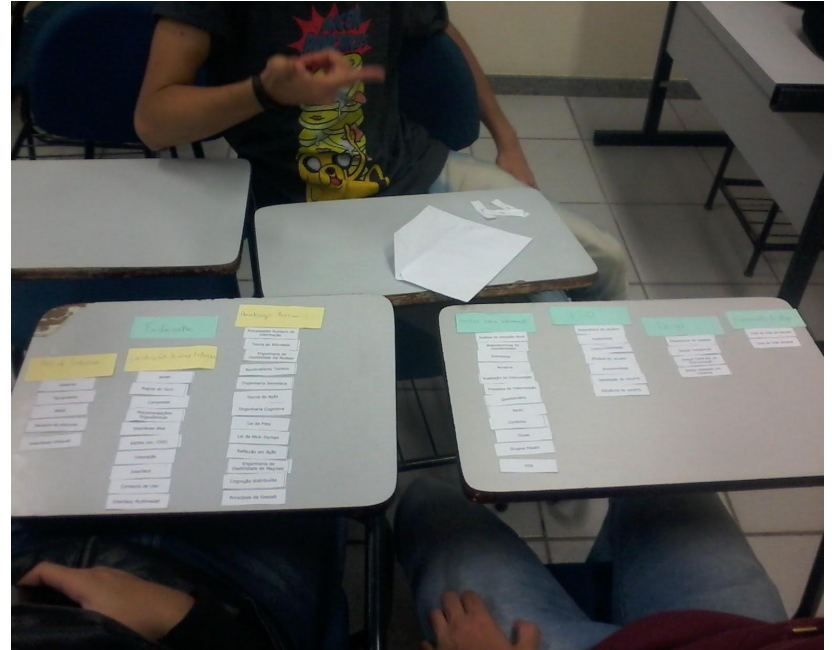
Information Architecture

In a cardsorting dynamic, students

- **organise** HCI topics into groups that make sense to them
- **label** the groups, forming categories of HCI topics

Insights for the information architecture in next versions of the support card

The card is renewed with each HCI course



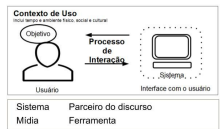
Cartão de Suporte: Interação Humano-Computador (IHC)



Prof. Lesandro Ponciano - lesandrop@pucminas.br
Departamento de Engenharia de Software e Sistemas de Informação

Conceitos Básicos

IHC trata desde os **arababuços técnicos** que guiam o **projeto, implementação e avaliação** de sistemas interativos até a compreensão dos **mais diversos fenômenos associados aos seus usos**.

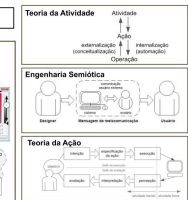
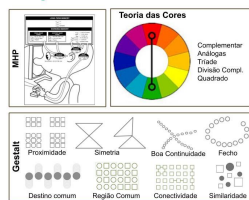


interface

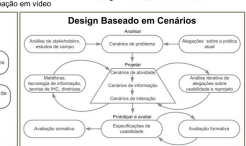
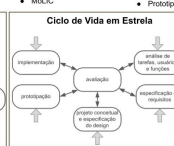
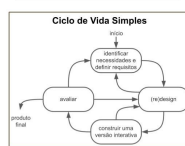
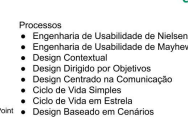
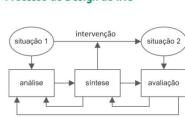
Affordance



Abordagens Técnicas



Processo de Design de IHC



Avaliação de IHC

- Inspeção**
- Avaliação Heurística
 - Método de Avaliação do Percorso Cognitivo (cognitive walkthrough)
 - Método de Inspeção Semiótica (MIS)

- Observação**
- Teste de Usabilidade
 - Método e Avaliação de Comunicabilidade (MAC)
 - Avaliação em protótipo em papel

- Investigação**
- Questionário System Usability Scale (SUS)
 - Entrevistas
 - Grupos focais

Cartão de Suporte: Interação Humano-Computador (IHC)



Curso Engenharia de Software - unidade Praça da Liberdade
Curso Sistemas de Informação - unidades Barreiro e Contagem

Regras e Princípios

Oito regras de ouro de Ben Shneiderman

1. Perseguir a Consistência
2. Fornecer Atalhos
3. Fornecer Feedbacks Informativos
4. Marcar o Fim dos Diálogos
5. Fornecer Prevenção de Erros e Formas Simples de Corrigir-los
6. Permitir Reversão de Ações
7. Fornecer Controle de Inicializa do Usuário
8. Reduzir a Carga de Memória de Trabalho

Sete princípios de Donald Norman

1. Usar ao mesmo tempo o conhecimento no mundo e o conhecimento na cabeça
2. Simplificar a estrutura dos menus
3. Tornar as coisas visíveis: encerrar ou sustentar as lacunas de execução e avaliação
4. Fazer constantemente os mapeamentos e as associações
5. Explorar o poder das coerções naturais e das artificiais
6. Projetar para o erro
7. Quando tudo o mais falhar, padronizar

Web Content Accessibility Guidelines (WCAG) 2.0

- 1.1 Fornecer alternativas em texto para todo o conteúdo não textual
- 1.2 Fornecer alternativas para multimídia dinâmica ou temporal
- 1.3 Criar conteúdo que possa ser apresentado de diferentes formas sem perder informação ou estrutura.
- 1.4 Facilitar aos usuários a audição e a visão dos conteúdos nomeadamente através da separação do primeiro plano do plano de fundo.
- 2.1 Fazer com que toda a funcionalidade fique disponível a partir do teclado
- 2.2 Proporcionar aos usuários tempo suficiente para lerem e utilizarem o conteúdo
- 2.3 Não criar conteúdo de uma forma que se sabe que pode causar convulsões
- 2.4 Fornecer formas de ajudar os usuários a navegar, localizar conteúdos e determinar o local onde estão.
- 3.1 Tornar o conteúdo textual legível e compreensível
- 3.2 Fazer com que as páginas da Web apareçam e funcionem de forma previsível
- 3.3 Ajudar os usuários a evitar e a corrigir os erros
- 4.1 Maximizar a compatibilidade com os agentes de utilizador atuais e futuros, incluindo as tecnologias de apoio

Recomendações Ergonômicas

Linguagem de comandos

- a. Manter um formato consistente
 - b. Usar nomes específicos, distintos e com significado
 - c. Permitir auto-completar
 - d. Gerar uma lista com diversos nomes possíveis e averiguar quais conferem maior significado
- Formulários**
- a. Diferenciação visual entre campos e dados
 - b. Diferenciação visual entre campos default e dados definidos pelo usuário
 - c. Diferenciação visual entre campos opcionais e campos de preenchimento obrigatório
 - d. Provêr feedback sobre quando os dados fornecidos são salvos (enviados) e se alguma ação é necessária ou não
 - e. Ordenar o Tab

Alertas

- a. Não tirar a atenção do usuário da tarefa principal, em algumas situações podem ser ignorados e em outros não podem
- b. Não sobreexpor o usuário com informações que ele não pode entender ou responder
- c. Cuidar da estrutura das mensagens que é apresentada
- d. Evitar linguagem ameaçadora ou alarmante (ex: "Erro fatal", "Execução abortada", "Deeser matar trabalho?", "Ocorreu um erro catastrófico!")
- e. Não usar duplas negativas, pois podem ser ambíguas
- f. O sistema deve "assustar a culpa" por erros
- g. Não use mensagens escritas em CAIXA ALTA, pois parece que se está gritando com o usuário
- h. Apresentar a mensagem por tempo suficiente para que ela seja percebida e peça confirmação se apropriado

Ícones

- William Horton Checklist
- a. **Conceitualizar:** A imagem sugere espontaneamente o conceito pretendido?
 - b. **Facilitar:** Os objetos nos ícones são familiares aos usuários?
 - c. **Conceitualizar:** A imagem está associada a apenas um conceito?
 - d. **Memorizar:** As operações estão bem descritas como objetos?
 - e. **Informativo:** Por que o ícone é importante?
 - f. **Diagnóstico:** O número de ícones é inferior a 20?
 - g. **Distintos:** Todos os ícones são diferentes uns dos outros?
 - h. **Distintos:** A imagem usa linhas suaves e evita contornos vivos?
 - i. **Legível:** Você lê todas as combinações de cor e tamanho em que o ícone será exibido?
 - j. **Consistência:** É cada objeto, cada ícone, cada pixel no ícone necessário?
 - k. **Consistência:** Está claro onde um ícone termina e outro começa?
 - l. **Extensível:** Possui design a imagem menor? Será que as pessoas ainda reconhecerão o ícone?

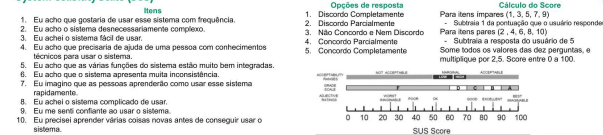
Avaliação Heurística



Etiquetas do Método de Avaliação da Comunicabilidade (MAC)

Cadê?	O que é isto?	Onde estou?	Por que não funciona?	Vai de outro jeito	Para mim está bom	Deixei
E agora?	Epa?	Ué, o que houve?	Assim não dá	Não, obrigadô!	Socorro	

System Usability Scale (SUS)



Considerações

Este cartão é um resumo de alguns dos tópicos abordados na disciplina IHC lecionada pelo Prof. Lesandro Ponciano, na PUC Minas. Trata-se de um material complementar e de consulta rápida durante as aulas. Não pode e não deve ser usado como única fonte de estudo para as avaliações da disciplina. Não pode ser usado durante avaliações em sala.

Referências

ROGA, Simone J. J. SILVA, Bruno Santana da. Interação humano-computador. Elsevier, 2010
ROGERS, Norman. SHARF, James. FREED, Andrew. Design de interação: além da interação homem-computador. ed. 2013
BENYON, David. Interação Humano-Computador - 2ª edição. Pearson

Materials and Methods

- The support card was created in 2018 and used in HCI classes at PUC Minas during the first semester of 2019
 - Software Engineering
 - Information Systems
- Observation and survey instrument
 - teacher's class notes
 - questionnaire answers (sample of 37 students)

Empirical Results

- Card used in different formats
 - Card on digital format (53%), printed on paper (33%), both formats (14%)
- Usefulness in the HCI classes
 - *“The card serves as a basic guide to remember the topics”*
 - *“It helps me in connecting the idea to the name”*
 - *“It is useful to organise content in my mind”*
- Usefulness beyond HCI classes
 - Interdisciplinary Software Work IV, Software Development Laboratory, Completion of Course Work, Software Testing
- All students would recommend the card to other HCI students

Conclusions...

- The card helps students in
 1. following the lessons in a global point of view
 2. integrating the different topics of the course
 3. using HCI topics in other classes
- It stimulates students' curiosity and engagement with HCI topics
 - input for students to build cognitive, mind, and concept maps
- In future work, we will study the use of support cards in the human computation domain

Thank you!

If you are interested, read our paper :)



Lesandro Ponciano

@lesandrop

lesandrop@pucminas.br



PUC Minas

Workshop on HCI Education, XVIII Brazilian Symposium on Human Factors in Computing Systems,
October 21-25, 2019 Vitória, ES - Brazil