Participant ID: _____

Software: Fusion 360 PowerPoint Python (with VSCode)

Slot 1:

Condition:	A – think-aloud	B – traditional-doc
Fusion:	Kitchen appliance	Piece of furniture
PowerPoint:	Historical event/period	Activity you enjoy
Python:	Connect four	Tic-tac-toe

Slot 2:

Condition:	A – think-aloud	B – traditional-doc
Fusion:	Kitchen appliance	Piece of furniture
PowerPoint:	Historical event/period	Activity you enjoy
Python:	Connect four	Tic-tac-toe

Facilitator setup

- AV stuff (video camera for recording participant screen; phone for recording interview audio)
- Stopwatch/timer (time each chunk to take 30 min)
- Giftcards
- Primary laptop
    - Fusion 360, PowerPoint, or VS Code
    - Word doc and Lightshot screenshot
    - Demo video
- Second laptop for Zoom call
- Print-outs for participants
    - knowledge_share_prompt.docx with sticky note “P” on back
    - shield_keywords.pptx with sticky note “W” on back
- Print-outs for facilitator
    - updated_final_study_script.docx
- Microphone
- Mouse

Script

Thanks for coming to participate in this study today. The study will last for 90 minutes. We're interested in understanding how people capture knowledge while they are working, and we have 2 different systems we'll have you work with today. You'll work with each of the systems for 30 minutes each, and at the end I will ask you a series of interview questions. Also, as we discussed over email, you will be using [Fusion/PowerPoint/Python]. I will present specific tasks to you shortly.

First I'd like for you to fill out this survey:…

Background for all tasks/conditions

As you work today, I'd like for you to imagine that you or your teammates will be revisiting your [code/slides/model] at some future date… for example, to make edits to it, to build on it, to repurpose it for another use case, to use it as an example to learn from, or to review it for quality. For each task I will be asking you to share knowledge that you think would be useful for you or your teammates to be reference. You can share anything you'd like, and as much/little as you'd like, but some particular items you may consider:

- **Design intent:** what your goals are, why you're doing what you're doing, the reason behind a design decision
- **Process:** how to use a tool or code library, best practices, suggestions on how to do something
- **Important:** anything you think is particularly important to know
- **To-do item:** something you or someone else needs to finish later
- **Problem:** issue you're experiencing

Condition A - Think-aloud tool

For this 30 min, I'll be asking you to work on a [coding/modeling/slide-building] task, and to speak as you do so. As I mentioned earlier, share information that you think would be useful to your teammates or future self, including but not limited to design intent, process, anything "important", to-do items, or problems you encounter.

Let me now demonstrate a tool you'll be using to help you think-aloud.
[Play the demo video I recorded]
[Flip over the "W" sheet, to show the shield widget]

IMPORTANT

- Make sure the participant does authentication first, before doing anything else; otherwise videos won't appear in live archive window
- Put live archive window in bottom left corner of screen so it's not a distraction, but needs to remain visible (in order for audio to be transcribed)

Think-aloud participant practice

- Have the participant practice talking
- Have them say keywords that correspond to some label/flag
- Also have them practice manually labeling

Actual task

IMPORTANT

- Start timer/stopwatch for 25min
- Start video camera recording
- Before they start 25 min task, make sure to click "start recording"

Condition B - Traditional documentation

For this 30 min, I'll be asking you to work on a [coding/modeling/slide-building] task, and to share knowledge in this Word document [and/or by writing comments in your code]. Here is also a screenshotting tool if you wish to include screenshots of your [model/code/slides].
[Demo the screenshotting tool – in demo video or TeamViewer;
Go to Mac bar at top and click the "feather" icon]

As I mentioned earlier, share information that you think would be useful to your teammates or future self, including but not limited to design intent, process, anything "important", to-do items, or problems you encounter.

IMPORTANT

- Start timer/stopwatch for 25min
- Start video camera recording

Fusion 360

Task 1
We'd like for you to create a **kitchen appliance** of your choosing, e.g., microwave, stove, sink, refrigerator.

Task 2
We'd like for you to create a piece of **furniture** of your choosing, e.g., chair, table, wardrobe

Please choose whichever task you are most comfortable with. Also, you don't need to learn any new features or workflows for this study; whatever knowledge you have already will be just fine :) But please feel free to search online for help if you need to, just as you would during your normal work day.

PowerPoint

Task 1
We'd like for you to make a PowerPoint on a **historical or current event/period** of your choosing, e.g., the moon landing, the Middle Ages, Raptors winning the NBA championships.

Task 2
We'd like for you to make a PowerPoint on an **activity/hobby you enjoy**, e.g., sports, music, woodshop/crafts, books, movies.

Please choose whatever task you are most comfortable with. Also, you don't need to learn any new software features for this study; whatever knowledge you have already will be just fine :) But please feel free to search online for help if you need to, just as you would during your normal work day.

Python in VS Code

Task 1
We'd like for you to create **tic-tac-toe** using Python. You can choose whatever interface you'd like for the user to interact with the game, either an actual GUI, or a text-based interface at the command line. You can start with an interface that lets 2 players take turns, and if there's time you could start making a basic AI for a single-player version.

Task 2
We'd like for you to create **connect-four** using Python. We'd like for you to create tic-tac-toe using Python. You can choose whatever interface you'd like for the user to interact with the game, either an actual GUI, or a text-based interface at the command line. You can start with an interface that lets 2 players take turns, and if there's time you could start making a basic AI for a single-player version.

You don't need to learn any new language constructs, technology, or algorithms for this study; whatever knowledge you have already will be just fine :) But please feel free to search online for help if you need to, just as you would during your normal work day.

For all tasks

Note that you will have 25 minutes in total for your task. Work at a pace that you normally would for your task. Don't worry too much about whether you completely finish it in the 25 min; just try your best. After the 25 min is up, I will give you instructions for a final 5 minute sub-task.

5-min review/reflect subtask

IMPORTANT:

- In think-aloud condition, first click "stop recording"
- Start timer/stopwatch for 5min

Reflect on your work and add anything you might have missed. Review the knowledge you captured, and make additions/edits as you see necessary, if you think anything needs to be changed.

- In think-aloud condition, point out all text editing features (edit existing utterances, and comments per utterance, per block, and ability to type or audio transcribe)

Notes

Think-aloud

Share post-study survey link; set as condition A

Traditional documentation

Share post-study survey link; set as condition B

Interview

IMPORTANT:

- Stop video camera recording
- Start interview audio recording (e.g., on phone)

Overall

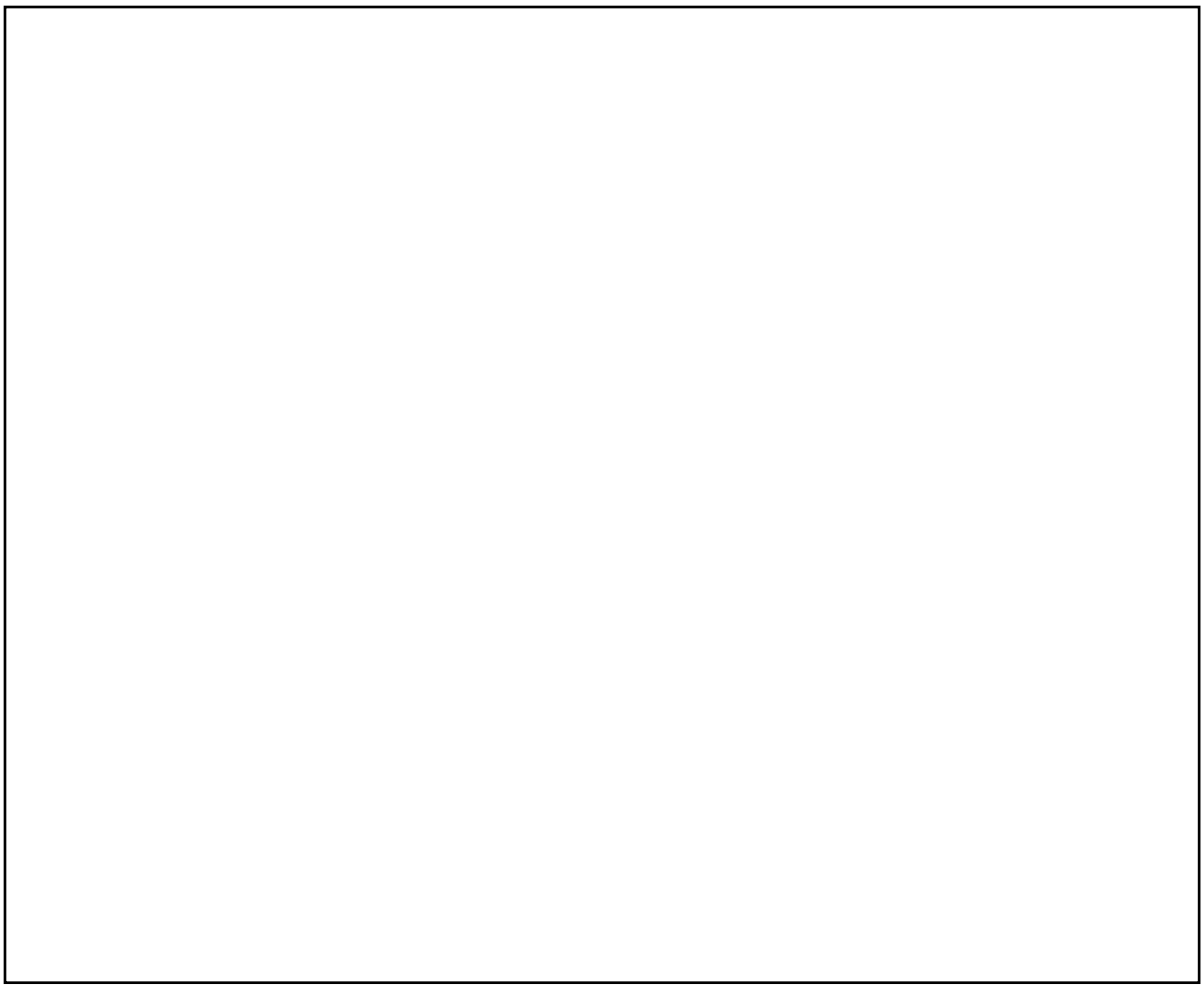
- What are your overall thoughts from this session?

Think-aloud

- What kinds of information did you find particularly useful or salient to share?

- When you decided to share information, what prompted you to do so? How did you decide what knowledge to share and when? What was your approach? How did you balance contributing information versus doing actual work?

- What did you think of the prompter/indicator widget?
    - How often did you find yourself looking at it?
    - Did you find the widget caused you to speak more?
    - Did you find it encouraged you to share more or specific kinds of information?
        - Or, was it more the fact that you were instructed to speak about these kinds of info (e.g., design intent, process) that encouraged you to speak about these?
    - Did you find it encouraged you to manually label utterances?
    - Was it informative?
    - Was it distracting?
    - Were there other types of labels that you think would be useful to mark for yourself, or for someone else who was picking up the tool later?

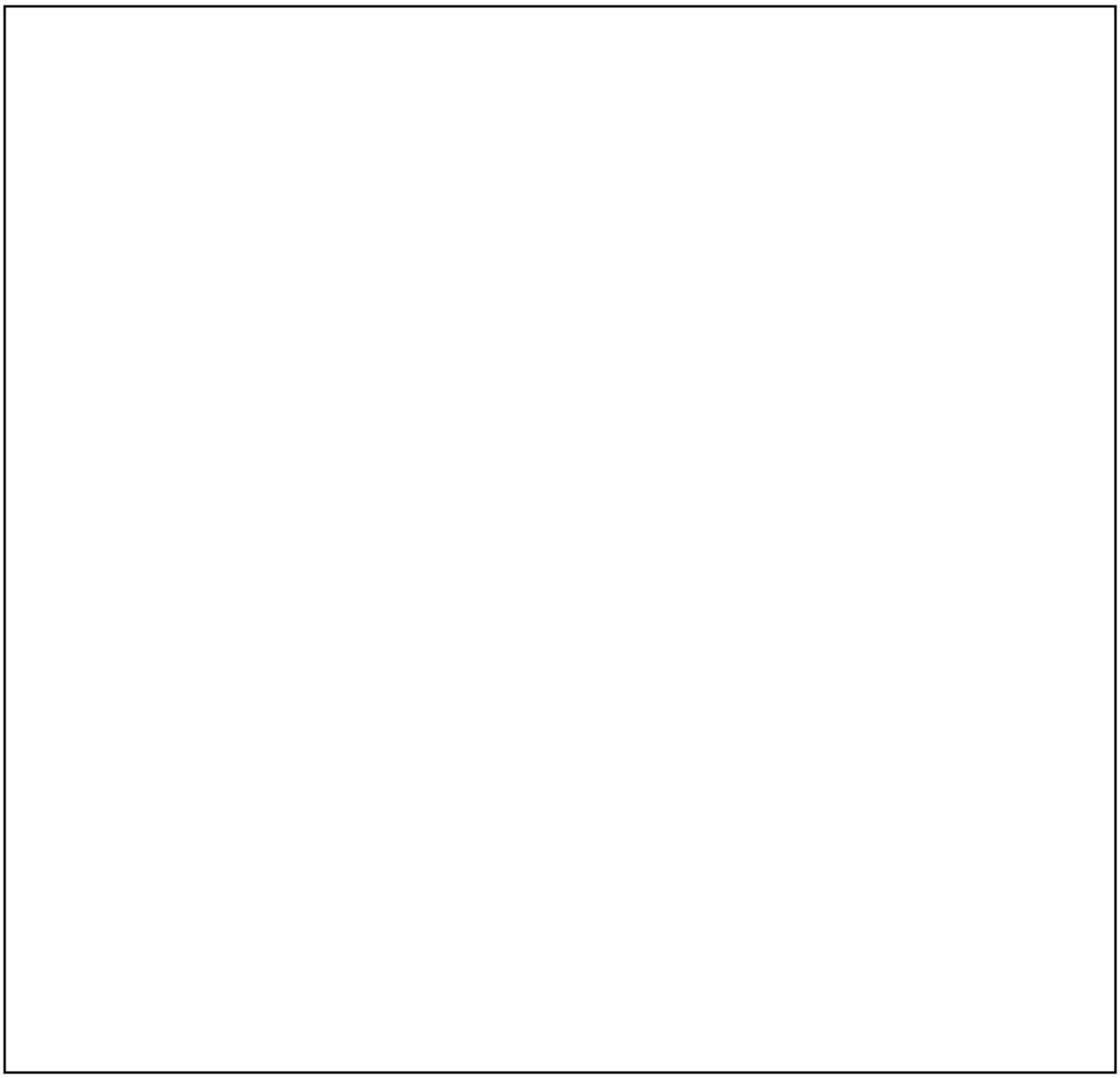


For Python participants ONLY

- Did you write code comments during the think-aloud condition? When and why? Does this reflect your usual process for code commenting?

- What did you think of the live archive window?
    - What did you think of the information captured there, and the way it was presented?
    - Did you add any comments afterwards to help clarify the recording or add information? Did you find yourself wanting to add comments to the utterances or blocks? Why did you add the comments that you did?
    - Was it easy to do these modifications/additions? Was the context provided by the embedded video and command pane helpful?
    - Thinking back to other projects that you've worked on, either your own projects, or projects or tasks that others worked on – do you think having this information would be helpful in understanding why certain decisions were made or why certain processes were used?

- What did you like or dislike about this knowledge capture paradigm?
    - Were there any aspects that worked particularly well for you and your workflow?
    - Were there any aspects that were challenging for you and your workflow?



Traditional documentation

- What did you think of capturing knowledge using the Word document and screenshot tool (and code commenting)?

- What did you like or dislike about this knowledge capture paradigm?
    - Were there any aspects that worked particularly well for you and your workflow?
    - Were there any aspects that were challenging for you and your workflow?

For Python participants ONLY

- What was your approach for writing in the Word doc versus writing code comments in the editor?

Overall

- Did you find capturing knowledge while working was distracting or impacted your productivity or performance?

- Did you find capturing knowledge while working to benefit the quality of your work?

- What are your thoughts on explicitly sharing info while working versus after working? Is there one approach you prefer overall, or that you prefer for specific scenarios?

- Who was the audience you were imagining for the information you were sharing? And what did you imagine their/your goals to be?

- Of the 2 options you tried out today (think-aloud vs Word doc/code comments), could you see yourself using one of them? Would they integrate well with your work process? Which do you prefer? Alternatively, is there some variation of one them you could see yourself using?

- ~~When do you think this type of information might be useful to yourself or others going forward?~~
- ~~Have you ever wished that you could have had this recorded information when looking at someone else's [model/slides/code], or your own [model/slides/code] many months after originally creating it?~~
- ~~Do you think you would use this approach (recording yourself for future benefit) going forward? Would it integrate well with your work process, or was it awkward? Do you have any suggestions on ways to make it more natural?~~

- Any other thoughts?

IMPORTANT:

- Stop audio recording

Debrief

- We're interested in capturing rich knowledge from workers, things that they're thinking in their heads but that is often unexpressed. People usually document important information through some form of writing (e.g., code comments, Word doc, email). However, these are distracting/tedious to do while working, and as a result may be brief.
- We believe that encouraging people to speak while they work, and/or at intervals, can result in rich knowledge capture, with hopefully minimal impact on productivity.
- We built this think-aloud system based on insights from a formative study we ran earlier in the summer, where we saw that people share a wide variety of information when speaking while working/after working.

Give participant giftcards

IMPORTANT: Gather all files

- Make sure they saved their work
    - Fusion
    - Python code
    - PowerPoint
    - Word doc
- Video camera task recording(s)
- Interview audio recording(s)