Pre-test interview

- What education do you have?
- What work experience do you have in programming?
- Which programming languages have you worked in?
- How much experience do you have in graphics programming?
  - Which APIs have you utilized?
    - Any experience with Vulkan or Direct3D 12?

Post-test interview

- What did you think of the tasks?
  - Level of difficulty?
- What's your first impression of the Papago API?
- Would you like to keep using Papago or use another API?
- What was good in Papago?
- What should have been better in Papago?
- Cognitive Dimensions Questionnaire:
  - Abstraction level
    - Did the API provide adequate control? Too much?
    - What was easiest to do?
    - What was most difficult?
  - Learning style
    - Would you be able to use this API without documentation?
    - Do you think you could learn it by exploration alone?
    - What was the most useful resource to learn the API?
  - Working framework
    - Did you have to think about too many different elements at once?
    - How many elements did you feel you needed to keep in mind during coding? Which elements?
  - Work-step unit
    - How much work did you have to do in order to enact a change in the program?
      - Was this a fitting amount of work?
  - Progressive evaluation
    - Did you feel that you could execute the application during coding in order to evaluate your progress?
    - What did you think of the feedback given through the API?
  - Premature commitment
    - What did you think of the dependency between API objects?
    - Did you ever have to make a decision before all information was available?
  - Penetrability
    - How easy was it to explorer the API features?
    - How easy was the API to understand?

- How did you go about retrieving what you needed to solve the tasks?
  - API elaboration
    - Any features, which were missing?
    - How would you expand upon the API?
  - API viscosity
    - How difficult was it to make changes to the system, when parts of it were already coded?
  - Consistency
    - After drawing a triangle, could you infer the remaining features of the API? How?
  - Role expressiveness
    - Was it ever difficult to tell what the role of different classes and methods were? Which?
  - Domain Correspondence
    - Were you previously familiar with the terminology of the API (commands, command buffers, queues etc.)?
    - Does the naming of these elements make sense?
    - Would you like to change any of the names? Which and why?