

# **Project: Keyboard-Based Iterative Zoom Navigation**

## **Project Goal:**

To develop and test a keyboard-only navigation system for selecting on-screen points.

## **Core Concept:**

The system allows a user to select a precise point by iteratively zooming in.

A grid of cells is overlaid on the target area.

This grid is mapped directly to a subset of keys on a QWERTY keyboard (we'll use the 24 keys that map to the row above and below the home row and the home row itself).

Pressing a key selects the corresponding cell.

The system then "zooms in" on that selected cell, which becomes the new target area, and a new grid is drawn.

This process repeats until the desired point is selected.

## **Core Hypothesis:**

This keyboard-centric method will be more efficient (faster and more accurate) than a traditional mouse by eliminating the time and cognitive load of switching between keyboard and mouse.

## **Development & Validation Plan:**

*Phase 1: Interactive Simulation:* Develop a basic simulation to allow for manual, direct interaction. This will be used for initial feasibility testing and "feel."

*Phase 2: Automated Trials:* If Phase 1 shows promise, automate the simulation to run a high number of trials. Each trial will involve selecting a random point in the target area and recording the "time" (e.g., number of keystrokes/steps).

*Phase 3: Analysis:* Use the data from the automated trials to validate the original hypothesis. This will involve comparing the system's performance against established benchmarks for mouse navigation (e.g., Fitts's Law).

*Phase 4: Documentation:* The findings will be documented in a formal paper.

*Phase 5: Production Build:* If the hypothesis is validated, we will proceed with building a full, production-ready solution.

# Overview

The application has two main panels stacked one on top of the other. A fixed image is shown in the bottom panel representing the target area. At the start of each “trial” the top area is grey until the user presses the space bar to start the exercise. A randomly located target, which moves for every trial, is then drawn on the bottom panel. Using only the keyboard, the user’s goal is to manoeuvre the tip of the pointer to within the target square. The application keeps track of key data for each trial such as the initial distance between the pointer and the centre of the target square and the number of key presses required to reach the target square. As soon as the tip of the pointer is within the boundary of the target square the trial concludes and the application present the data recorded.

Immediately below is an illustration of the main application before a trial starts. The following pages illustrate a single trial’s progression screen-by-screen. Note: these are illustrations to explain the process - the images are not pixel-perfect.

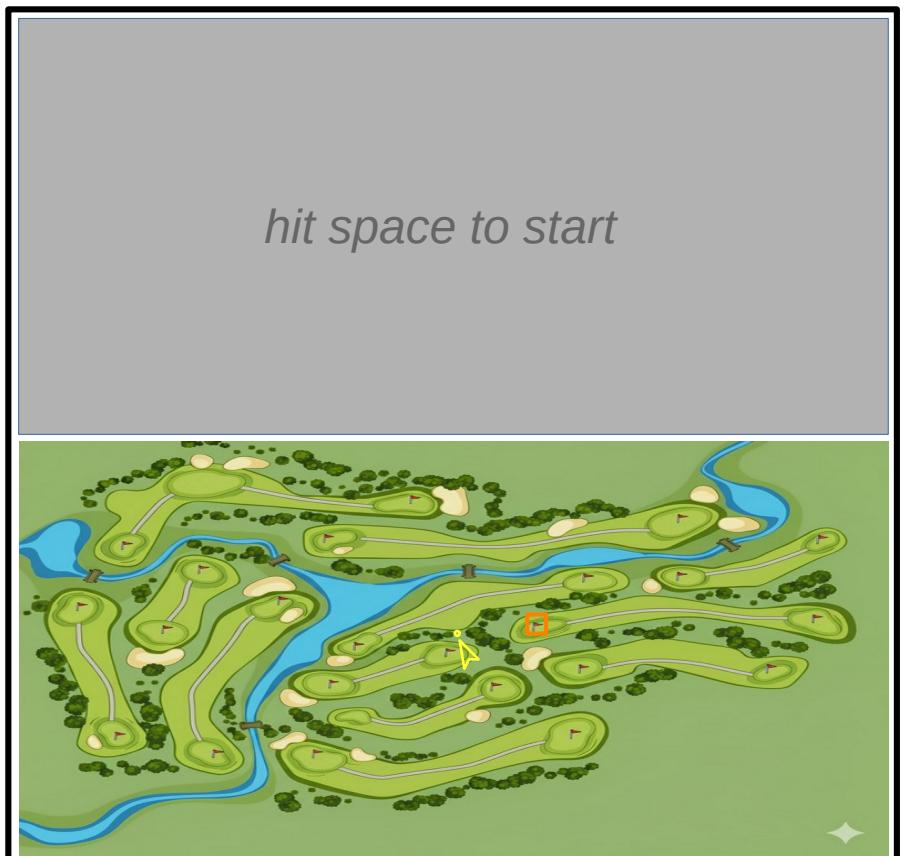
The top panel shows the grid overlaid on an image of the area it represents

*hit space to start*

The bottom panel represents the entire screen – a yellow pointer is initially located with its tip in the centre of the screen.

At the start of each run, a randomly chosen point will be identified with a an orange box in the target area panel.

The goal is to navigate tip of the pointer to within the box.



# Starting the trial

*Selecting the space key starts the trial...*

The top panel then shows a grid overlaid on an image of the area it represents



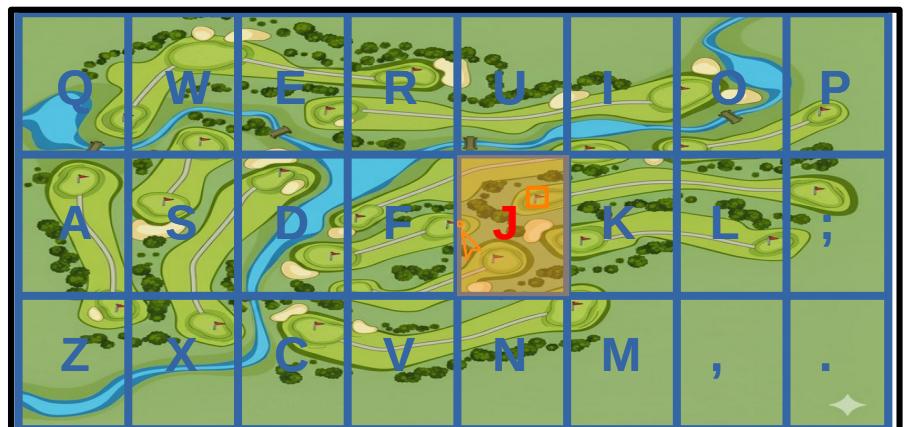
The bottom panel represents the entire screen – a red outline shows the area currently covered by the grid and an orange pointer is overlaid to show where the target pointer is currently.



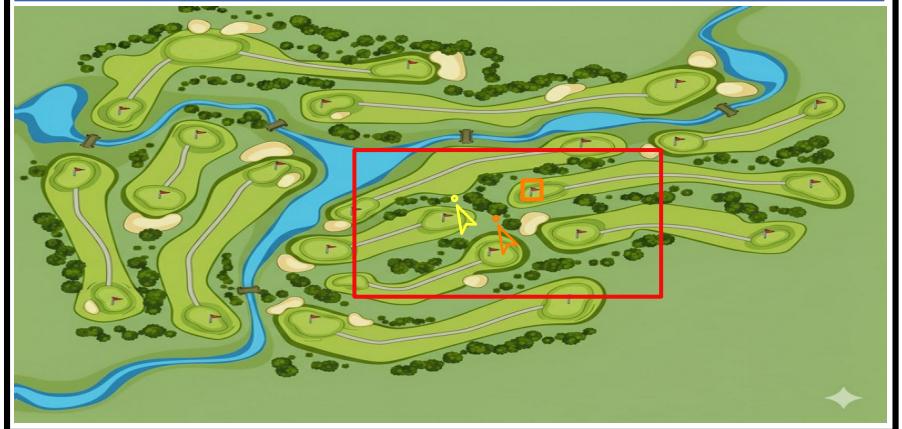
# First move initiated

The target is in the "J" cell so, the user hits the "J" key...

User selects the "J" cell by hitting the "J" key – the corresponding cell flashes with an orange background and the letter turns to red to confirm the selection.



An outline of the new target area flashes and then is drawn. The new area is centred on the middle of the area represented by the "J" cell and maintains the proportions of the grid. The orange pointer overlay is updated to show where the pointer will be moved to.



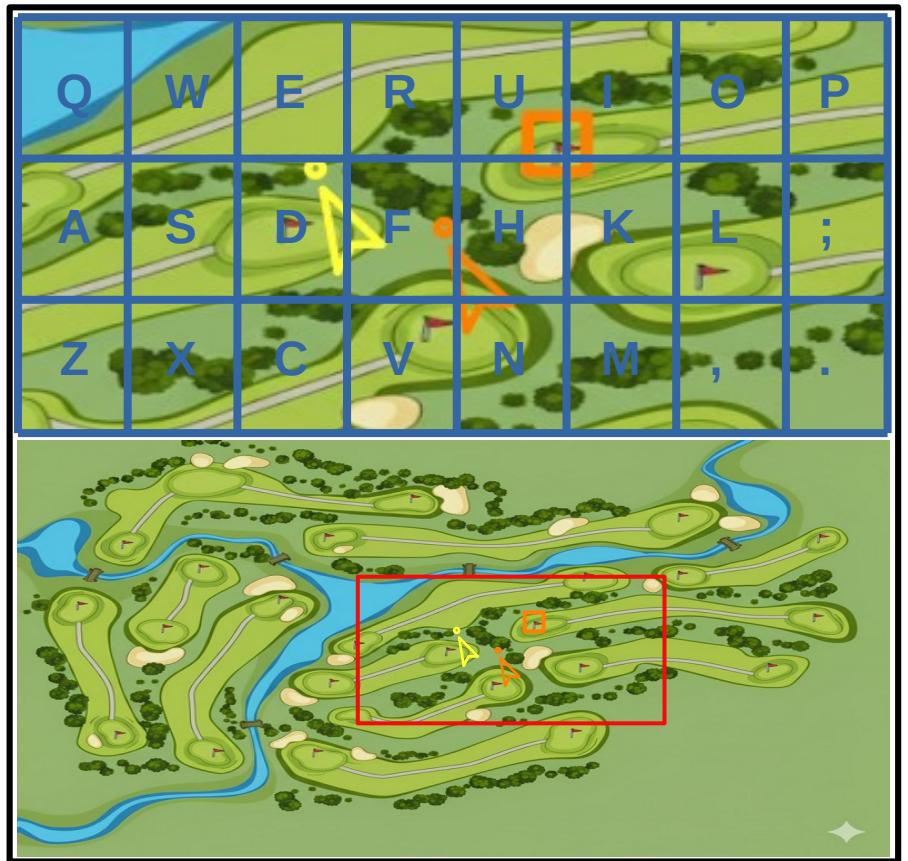
# First zoom completes

*After initial UI feedback, panels are updated to the new zoom level*

The image in the selection grid is updated to show the zoomed in area that is now highlighted by the target area outline.

Note that the boundary of the target is now straddling multiple cells.

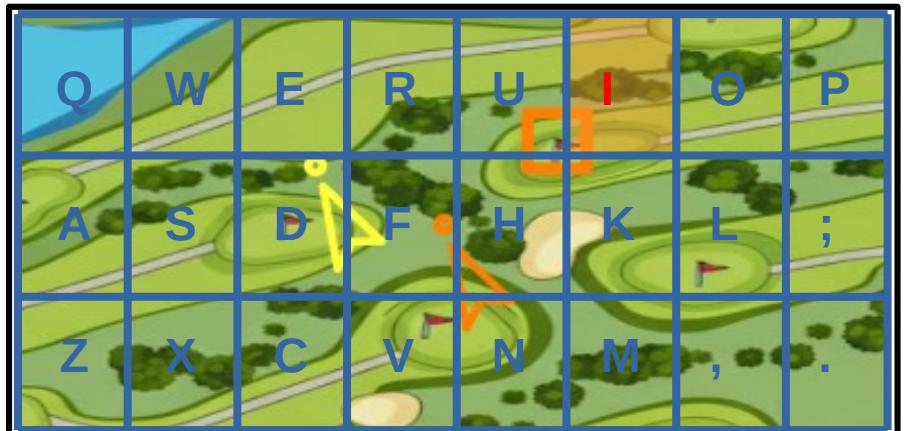
The tip of the target pointer, the one in orange, should always be on the centre of the grid between the "F" and "H" cells.



## Second move initiated

For the second move, the user could target any one of four cells but, since there is more of the area of the target in the "I" cell, that is the one the user chooses so, the user presses the "I" key...

User selects the "I" cell by pressing the "I" key – the corresponding cell flashes with an orange background and the letter turns to red to confirm the selection.



An outline of the new target area flashes and then is drawn. The new area is centred on the middle of the area represented by the "I" cell and maintains the proportions of the grid. The orange pointer overlay is updated to show where the pointer will be moved to.

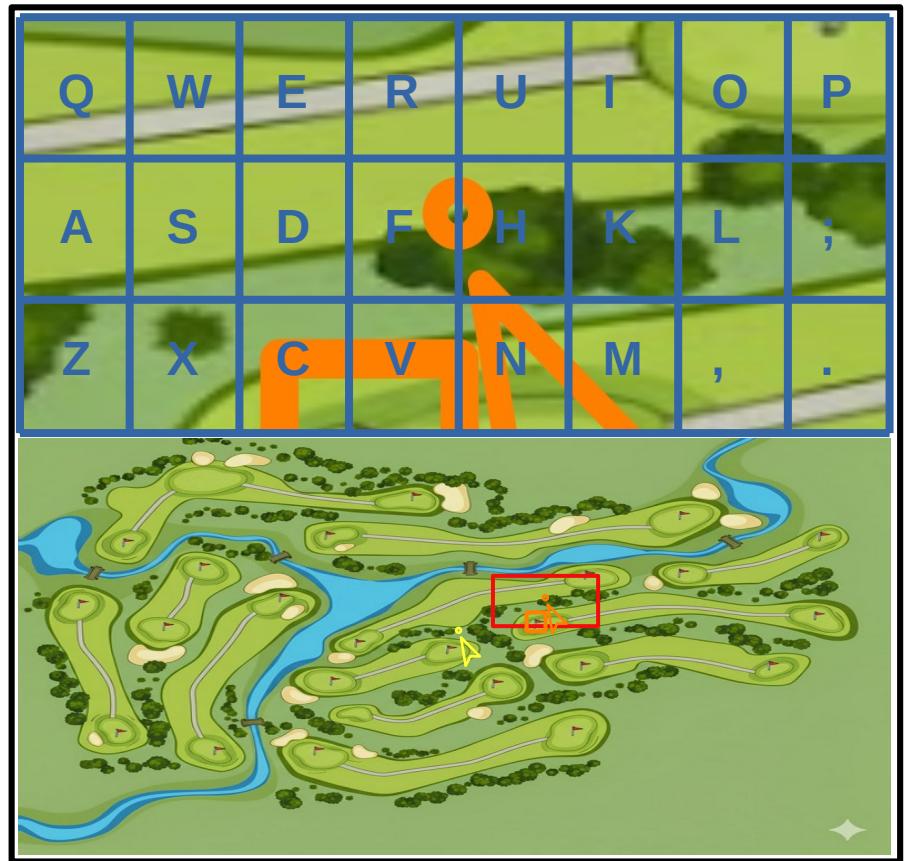


## Second zoom completes

*After initial UI feedback, panels are updated to the new zoom level*

The image in the selection grid is updated to show the zoomed in area that is now highlighted by the target area outline.

Note the target is now straddling the "C", "V" and (just) the "N" cells.



## Third move initiated

For the third move, the user could target any one of three cells but, since there is more of the area of the target in the “V” cell, that is the one the user chooses so, the user presses the “V” key...

User selects the “V” cell by pressing the “V” key – as before, the corresponding cell flashes with an orange background and the letter turns to red to confirm the selection.



Again, as with the previous move, an outline of the new target area first flashes in red and is then drawn on the lower panel. The new target area is centred on the middle of the area represented by the “V” cell.

The top and bottom of the new target area correspond to the top and bottom of the V cell. The left and right edges of the new target area are chosen to maintain the proportions of the grid.



## Third zoom completes

*After initial UI feedback, panels are updated to the new zoom level*

The image in the selection grid is updated to show the zoomed in area that is now highlighted by the target area outline.

Note the target is now straddling most of the bottom two rows of cells but, importantly, the centre of the target pointer is now within the target area.



## Target reached, trial completes.

*After initial UI feedback, panels are updated to the new zoom level – as the pointer is now in the target boundary the user completes the trial by pressing the space key.*

A new overlay is presented in the top panel summarising the trial.

The overlay shows the distance the pointer moved, as a percentage of the target area, and the number of key presses required to get there.

As the message shows, the user can then press the space key to start another trial.

The pointer moved 15% of the target area during the trial and took 3 moves.

*press space to start another trial.*

