THE UNIVERSITY OF ALABAMA
COMPUTER SCIENCE

WAVER Web-based Application for Virtual Exercise Regimen

Christina Noe

The Difficulties in Pursuing Rehabilitation

Physical therapy patients may face adversity in pursuing rehabilitation:

PHYSICAL & MENTAL

- Distanced far from medical facilities
- Mobility-impairing/painful disability
- Forced from the comfort of home

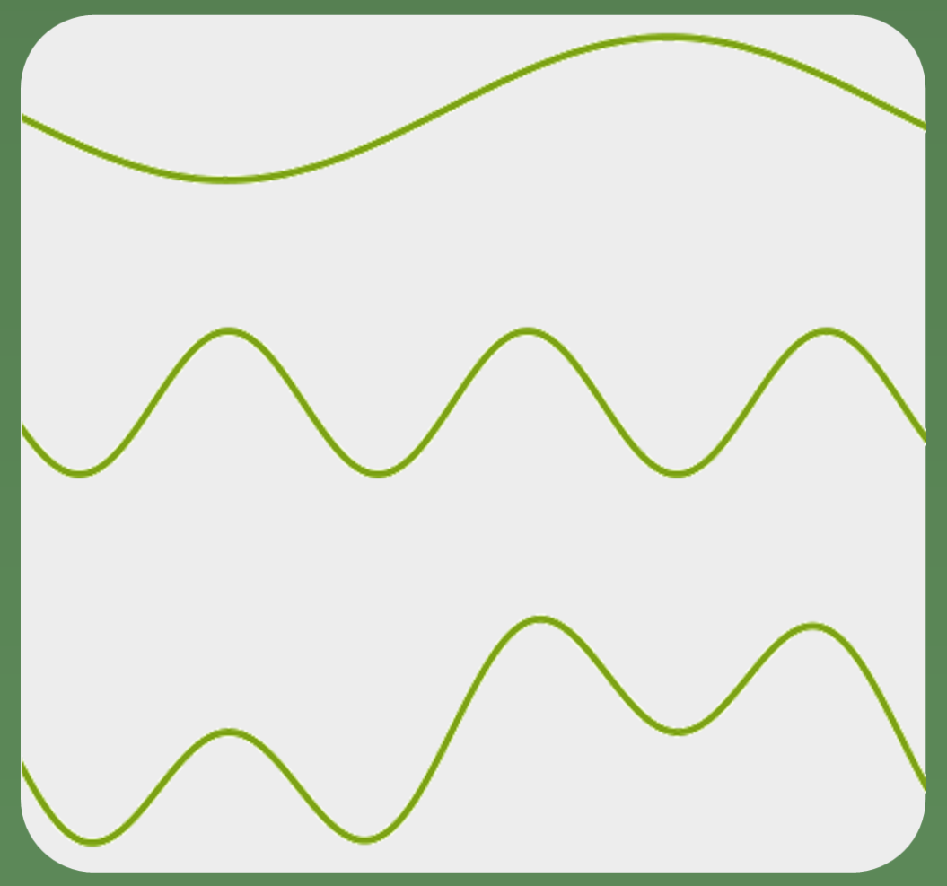
FINANCIAL

COST PER REHAB SESSION [1]:

With insurance	$10-75
Without insurance	$50-350

Task-oriented Movement

Research has shown that repetitive task-oriented movement is integral in upper-limb rehabilitation. In one study, individuals used their finger to track a sine wave on a computer screen [2]. These individuals exhibited greater tracking accuracy along with grasp and release function over individuals who had not yet received the same treatment.



[1] "How Much Does Physical Therapy Cost? - CostHelper.com." *CostHelper*. N.p., n.d. Web. 22 Feb. 2015.

[2] Carey, James R., Teresa J. Kimberley, Scott M. Lewis, Edward J. Auerbach, Lisa Dorsey, Peter Rundquist, and Kamil Ugurbil. "Analysis of FMRI and Finger Tracking Training in Subjects with Chronic Stroke." Brain 125.4 (2002): 773-88. Web.

3D LEAP Motion Device

The LEAP is a motion-capture hardware device that supports hand and finger motions as input. When a user waves their hand over the device, the motion is translated into an arbitrary action on-screen.

KEY FEATURES

Small size	(2" length, 1" width)
Inexpensive	(under $100)
Simple usage	(USB interface)

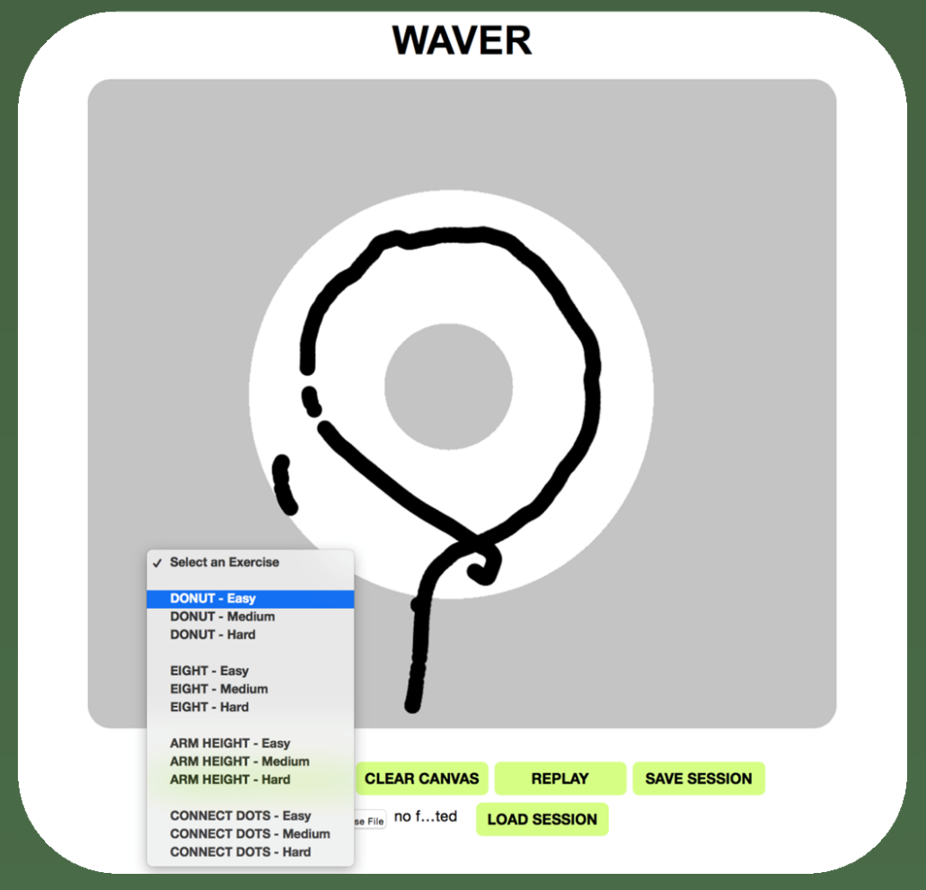
Solution: WAVER

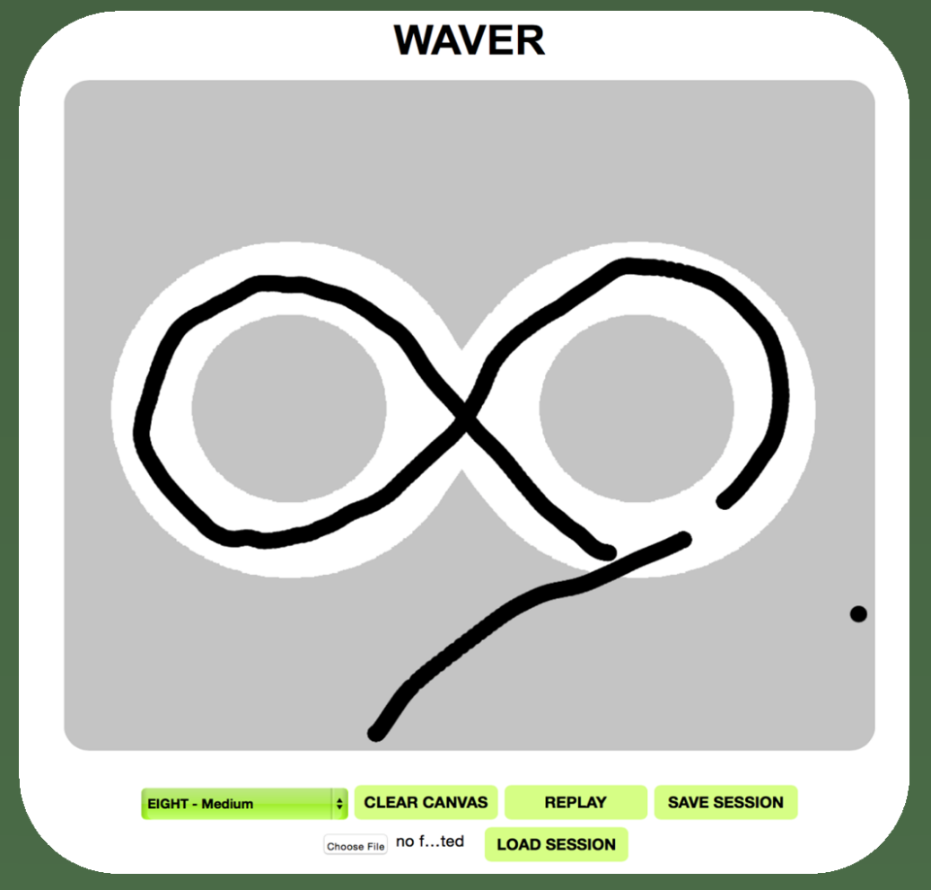
WAVER is a web application that utilizes the 3D LEAP motion device to assist patients who have upper-limb mobility and dexterity limitations.

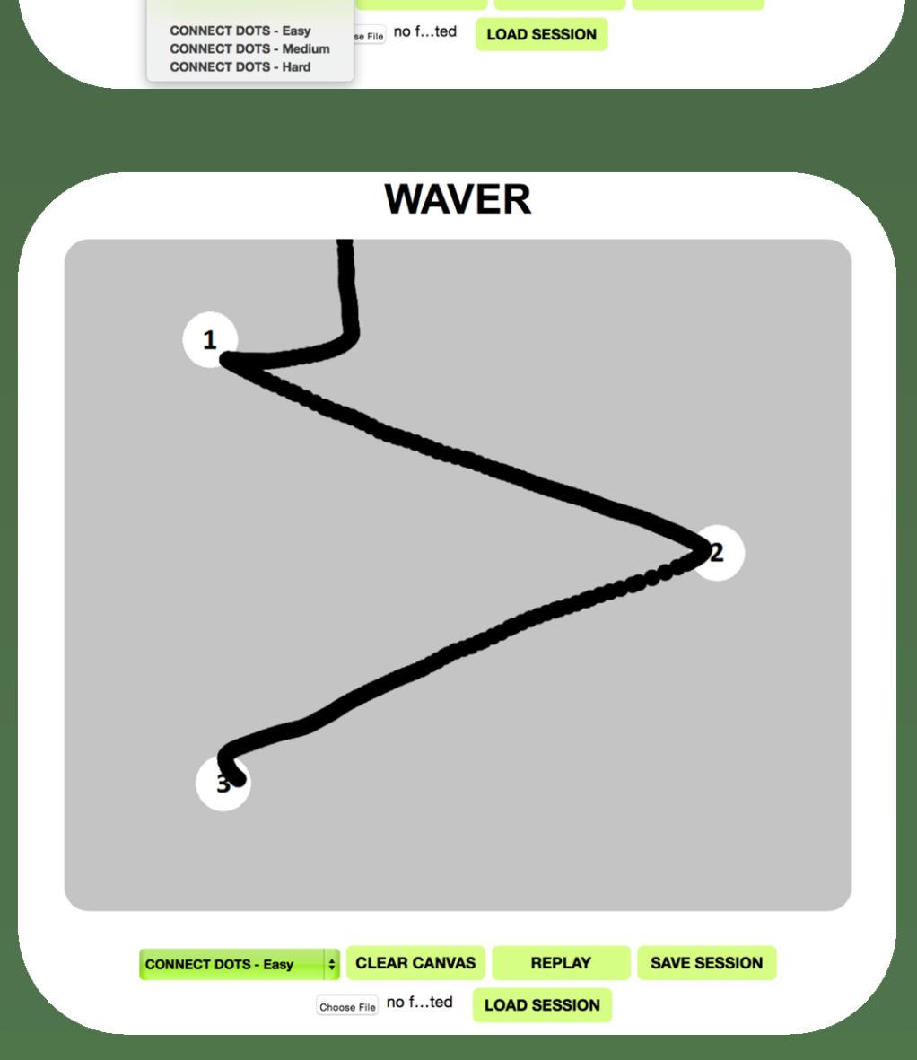
WAVER presents drawing exercises to a patient that can be performed on the patient's home computer. Drawings tracked by WAVER provide on-screen feedback to the patient, as well as the potential to share the results remotely with a therapist.

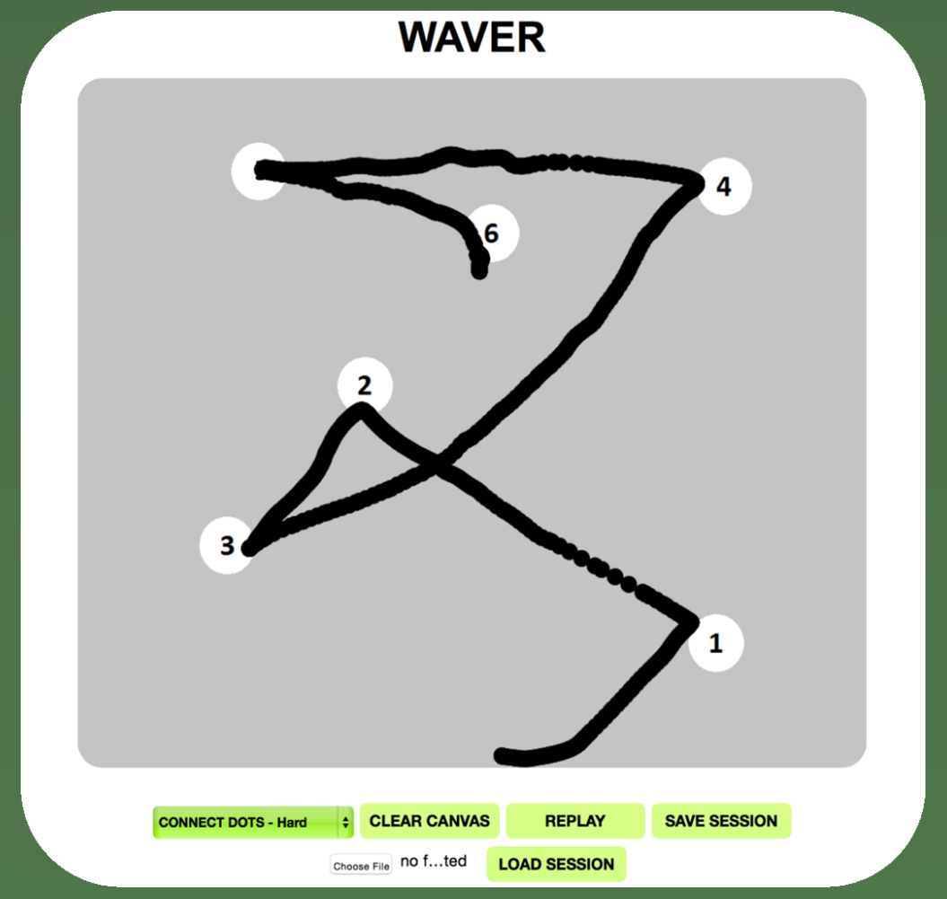
WAVER provides inexpensive, home-based rehabilitation for patients suffering through stroke, disability, or even sports-related injury.

WAVER in Action









PROCESS

1. User connects and installs LEAP device
2. An exercise is selected. User can see the motion of their hand tracked as a drawing on the screen in relation to the image suggested by WAVER.

KEY FEATURES

SESSION REPLAY:

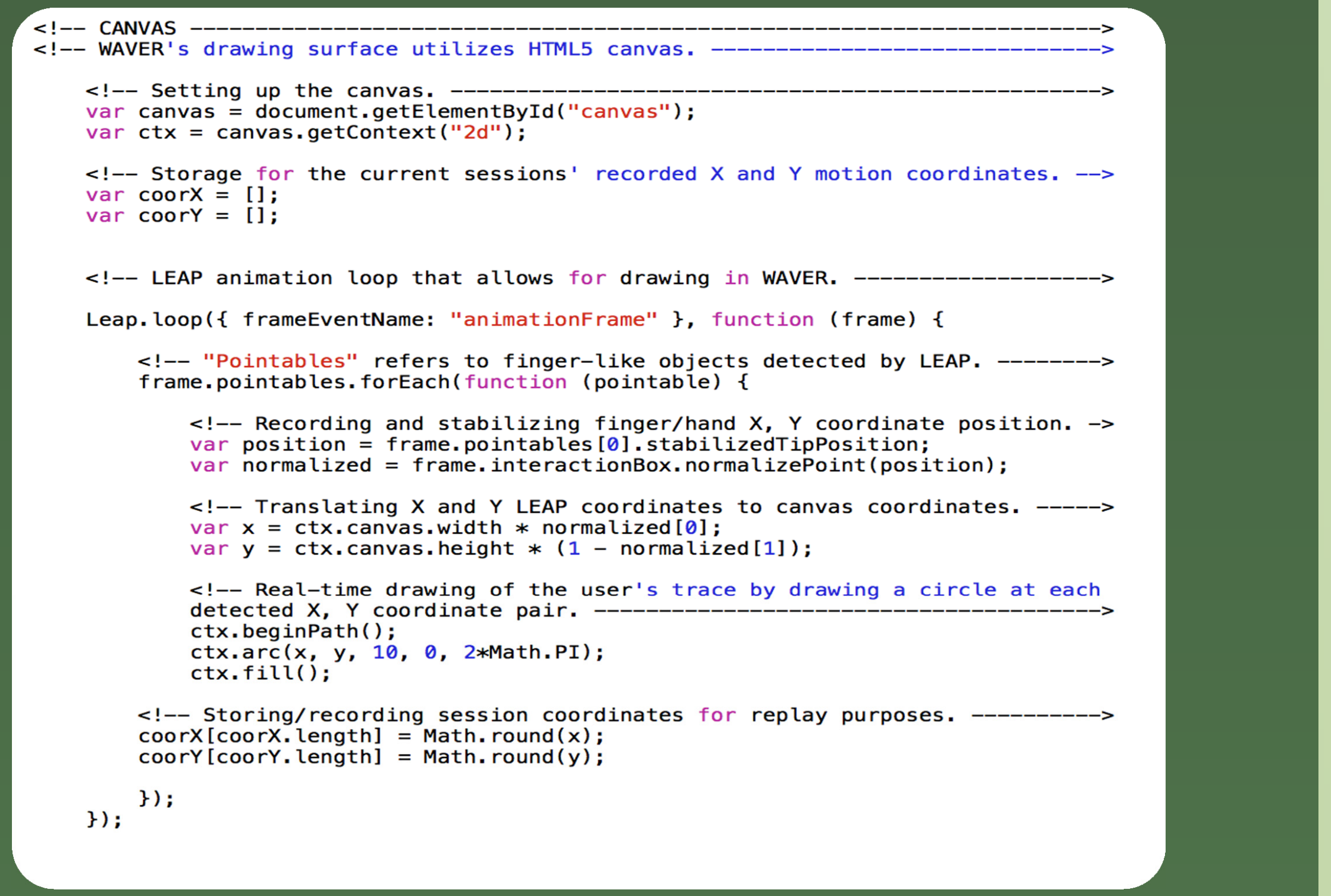
WAVER allows the user to replay their current session even after the canvas is reset. Users can also save sessions to their computer as a file containing a set of coordinates. Previous sessions can then be loaded back into WAVER and replayed.

ENGAGING IMPROVEMENT:

WAVER provides an engaging, encouraging visual measure of improvement for rehabilitation patients. Future versions will include the ability to send session results remotely to therapists to enable real-time interaction with a therapist.

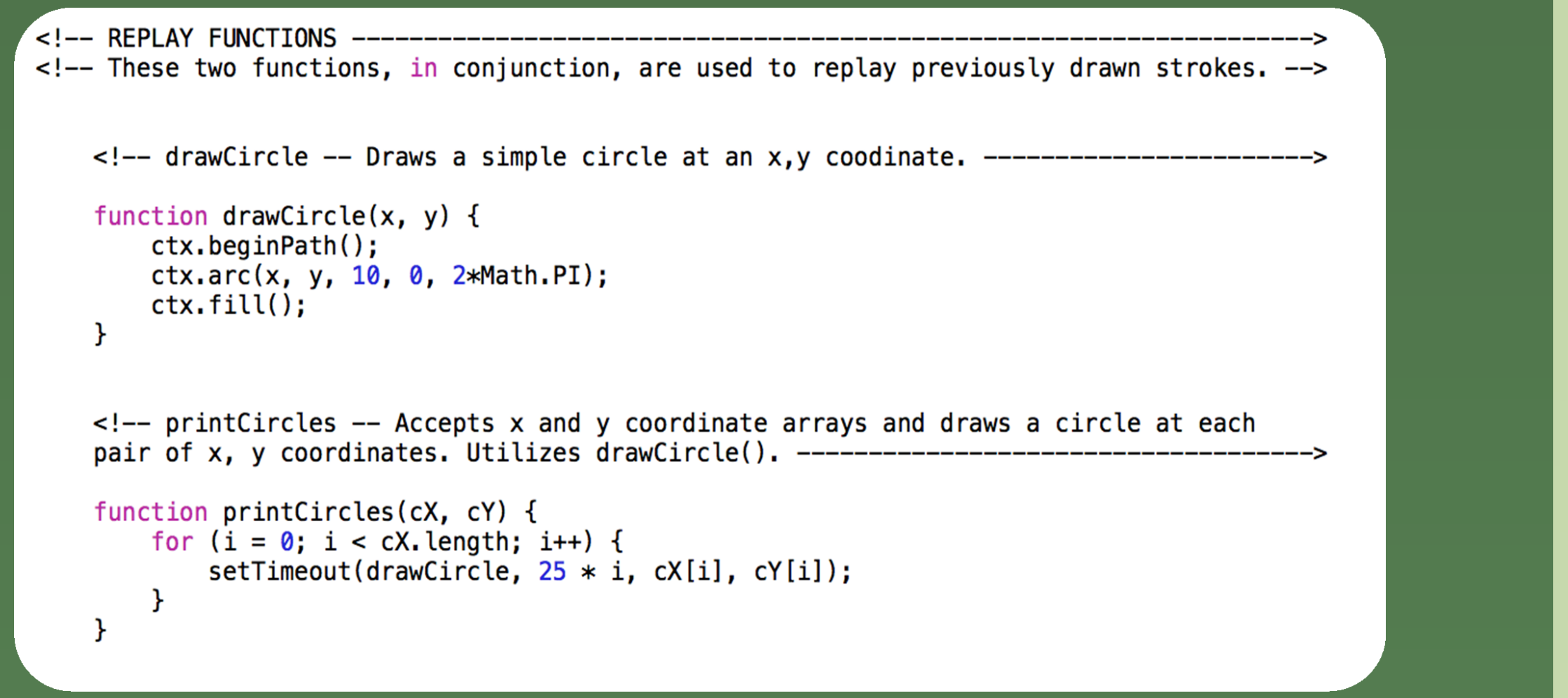
WAVER Code

HTML5 CANVAS



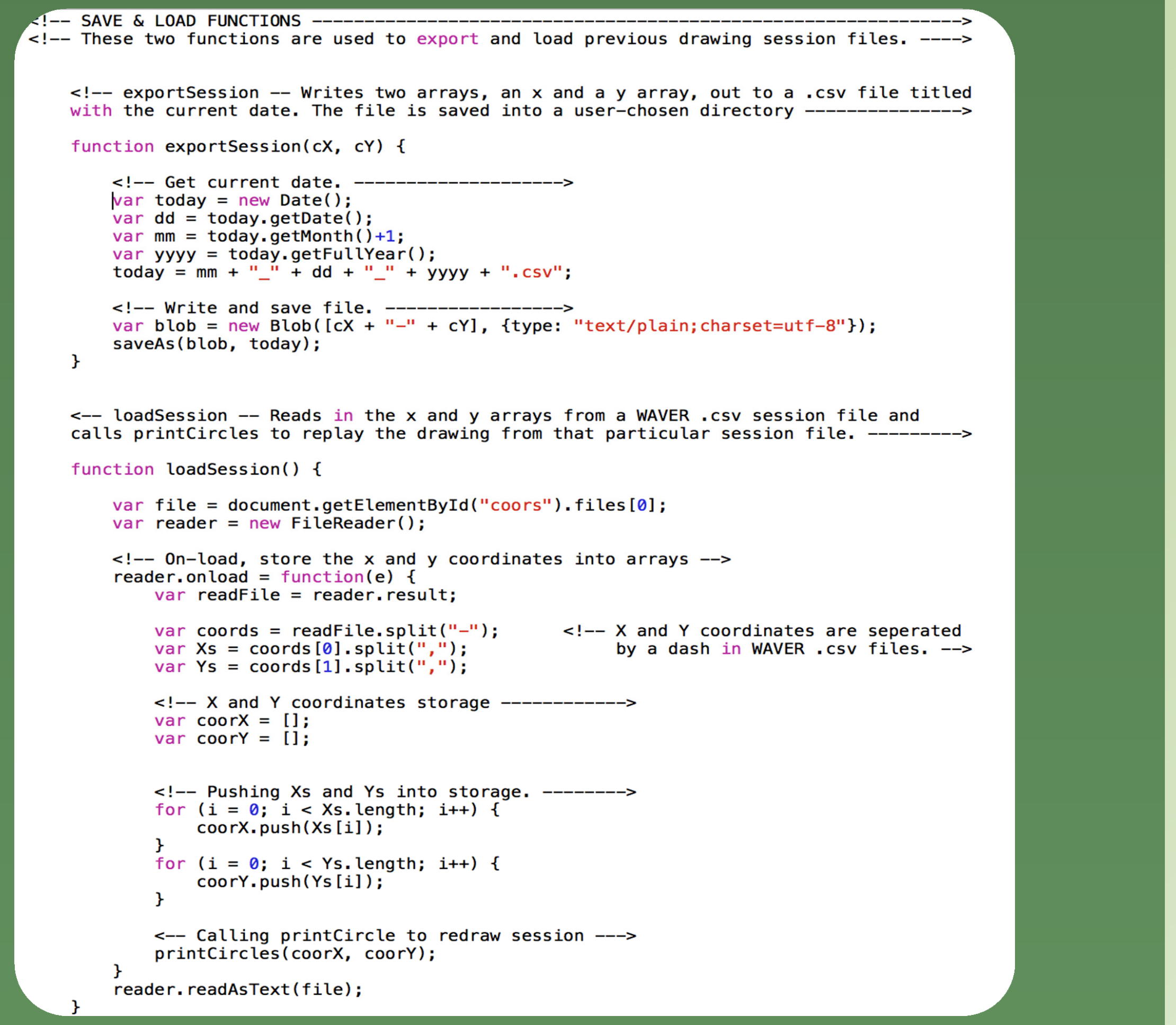
```
<!-- CANVAS ------------------------------------------------------------------->
<!-- WAVER's drawing surface utilizes HTML5 canvas. ----------------------------->

    <!-- Setting up the canvas. ---------------------------------------------->
    var canvas = document.getElementById("canvas");
    var ctx = canvas.getContext("2d");

    <!-- Storage for the current sessions' recorded X and Y motion coordinates. -->
    var coorX = [];
    var coorY = [];


    <!-- LEAP animation loop that allows for drawing in WAVER. ------------------->

    Leap.loop({ frameEventName: "animationFrame" }, function (frame) {

        <!-- "Pointables" refers to finger-like objects detected by LEAP. -------->
        frame.pointables.forEach(function (pointable) {

            <!-- Recording and stabilizing finger/hand X, Y coordinate position. ->
            var position = frame.pointables[0].stabilizedTipPosition;
            var normalized = frame.interactionBox.normalizePoint(position);

            <!-- Translating X and Y LEAP coordinates to canvas coordinates. ----->
            var x = ctx.canvas.width * normalized[0];
            var y = ctx.canvas.height * (1 - normalized[1]);

            <!-- Real-time drawing of the user's trace by drawing a circle at each
            detected X, Y coordinate pair. ---------------------------------------->
            ctx.beginPath();
            ctx.arc(x, y, 10, 0, 2*Math.PI);
            ctx.fill();

        <!-- Storing/recording session coordinates for replay purposes. ---------->
        coorX[coorX.length] = Math.round(x);
        coorY[coorY.length] = Math.round(y);

        });
    });
```

REPLAY SESSION



```
<!-- REPLAY FUNCTIONS --------------------------------------------------------->
<!-- These two functions, in conjunction, are used to replay previously drawn strokes. -->


    <!-- drawCircle -- Draws a simple circle at an x,y coodinate. ---------------------->

    function drawCircle(x, y) {
        ctx.beginPath();
        ctx.arc(x, y, 10, 0, 2*Math.PI);
        ctx.fill();
    }


    <!-- printCircles -- Accepts x and y coordinate arrays and draws a circle at each
    pair of x, y coordinates. Utilizes drawCircle(). ------------------------------->

    function printCircles(cX, cY) {
        for (i = 0; i < cX.length; i++) {
            setTimeout(drawCircle, 25 * i, cX[i], cY[i]);
        }
    }
```

SAVE & LOAD SESSION



```
<!-- SAVE & LOAD FUNCTIONS ------------------------------------------------------->
<!-- These two functions are used to export and load previous drawing session files. ---->


    <!-- exportSession -- Writes two arrays, an x and a y array, out to a .csv file titled
    with the current date. The file is saved into a user-chosen directory --------------->

    function exportSession(cX, cY) {

        <!-- Get current date. -------------------->
        var today = new Date();
        var dd = today.getDate();
        var mm = today.getMonth()+1;
        var yyyy = today.getFullYear();
        today = mm + "_" + dd + "_" + yyyy + ".csv";

        <!-- Write and save file. ----------------->
        var blob = new Blob([cX + "-" + cY], {type: "text/plain;charset=utf-8"});
        saveAs(blob, today);
    }


    <-- loadSession -- Reads in the x and y arrays from a WAVER .csv session file and
    calls printCircles to replay the drawing from that particular session file. --------->

    function loadSession() {

        var file = document.getElementById("coors").files[0];
        var reader = new FileReader();

        <!-- On-load, store the x and y coordinates into arrays -->
        reader.onload = function(e) {
            var readFile = reader.result;

            var coords = readFile.split("-");      <!-- X and Y coordinates are seperated
            var Xs = coords[0].split(",");              by a dash in WAVER .csv files. -->
            var Ys = coords[1].split(",");

            <!-- X and Y coordinates storage ------------>
            var coorX = [];
            var coorY = [];


            <!-- Pushing Xs and Ys into storage. -------->
            for (i = 0; i < Xs.length; i++) {
                coorX.push(Xs[i]);
            }
            for (i = 0; i < Ys.length; i++) {
                coorY.push(Ys[i]);
            }

            <-- Calling printCircle to redraw session --->
            printCircles(coorX, coorY);
        }
        reader.readAsText(file);
}
```