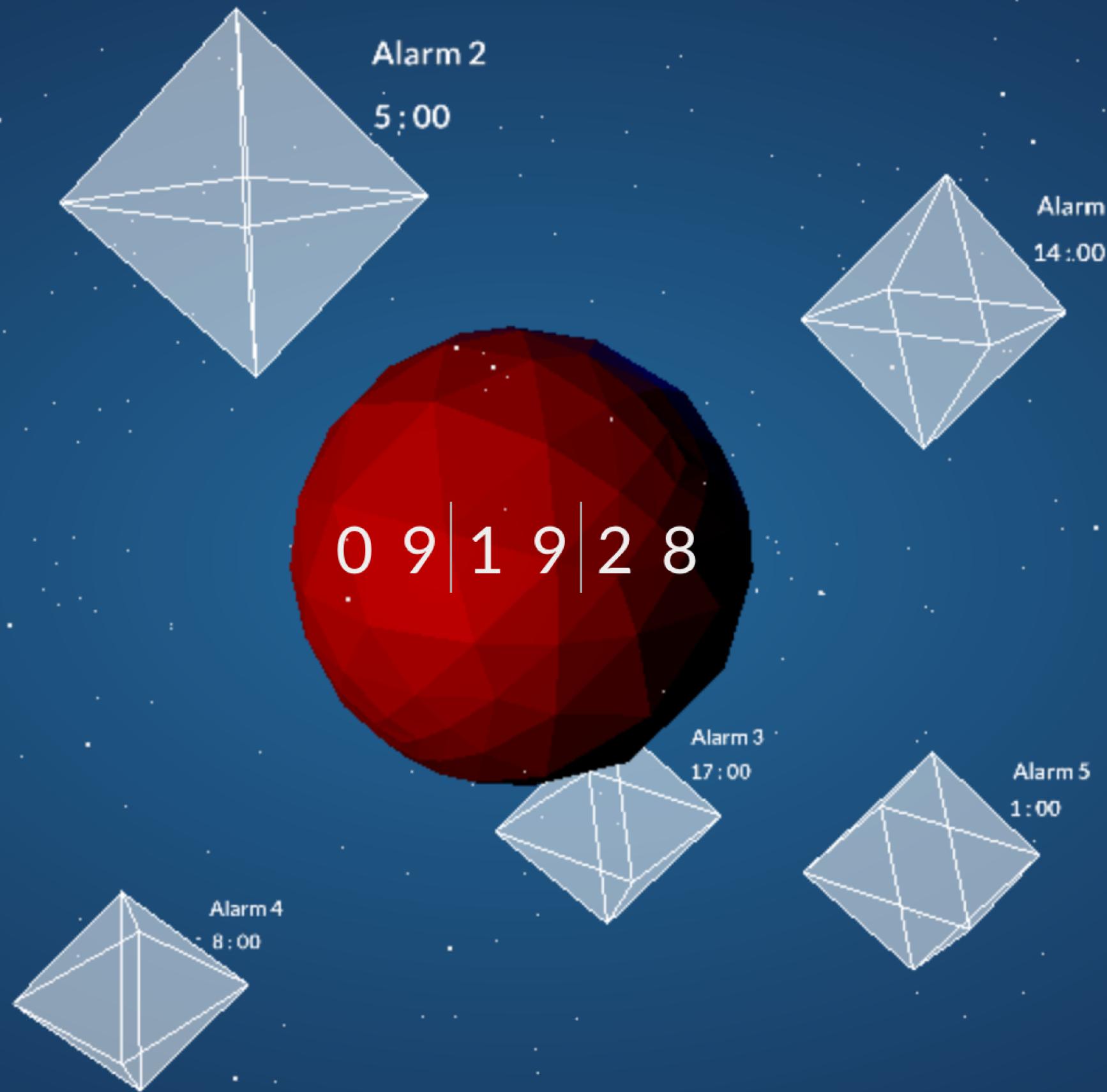


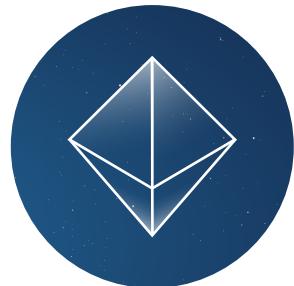
# IAT381 UX REDESIGN gem.

TEAM ANGULAR DUMMIES  
Kay Lin + Marya Smirnova



# ux features

## PLAYFUL 3D ALARM CLOCK



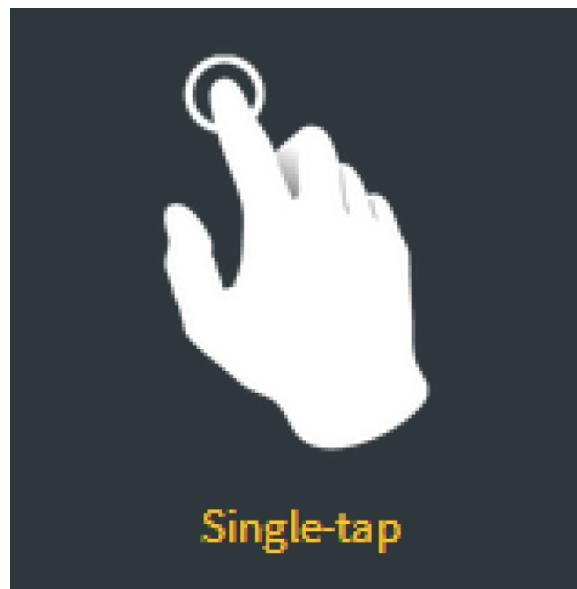
# ux features

## UNIQUE GESTURES



Pinch Out

**PINCH IN**  
to add an alarm



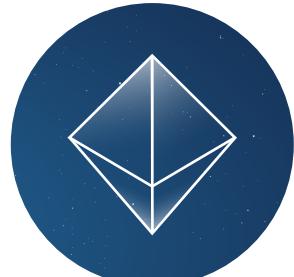
Single-tap

**SINGLE TAP**  
to select an alarm



Pinch

**PINCH OUT**  
to delete an alarm

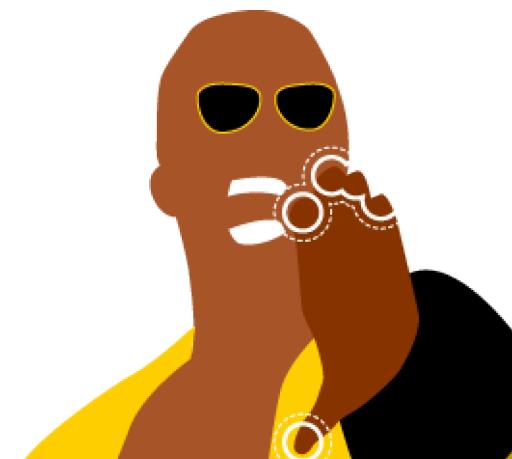


framework

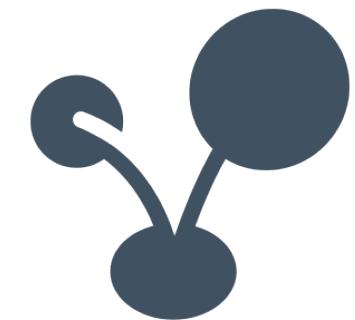


libraries

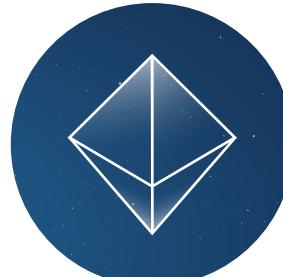
three.js



HAMMER JS



TWEENJS



```

function manageMultitouch(event) {

    if(event.type === 'tap') {

        if( event.pointerType === "touch") {
            editMode = true;
            alarmEdit = false;
            touchPos.x = ( event.pointers[0].clientX/ renderer.domElement.width ) * 2 - 1;
            touchPos.y = - ( event.pointers[0].clientY / renderer.domElement.height ) * 2 + 1;

            raycaster.setFromCamera( touchPos, camera );

            var intersects = raycaster.intersectObjects( targetList );

            if ( intersects.length > 0 ) {

                intersects[ 0 ].object.material.color.setHex( 'blue' );

                // console.log(intersects[0].object.id);
                k = targetList.indexOf(intersects[0].object.id);
                selectedObject = intersects[0].object;
                toDel= intersects[0].object;

                cameraX = selectedObject.position.x;
                cameraY = selectedObject.position.y;
                cameraZ = -selectedObject.position.z + 200;

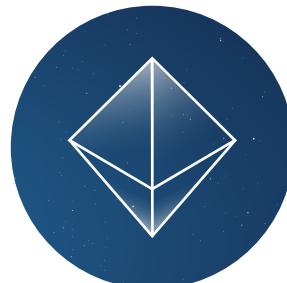
                // console.log(cameraX,cameraY,cameraZ);

                // document.getElementById("resultDIV").innerHTML = ("tap: "+ intersects[0].object.id);

            }
        }
    }
}

```

**custom logic**  
MANAGING MULTIPLE GESTURES



```

if(event.type === 'pinchin'){

    if( event.pointerType === "touch"){

        scene.remove( toDel );

        var tween2 = {
            scale: 0,
            go: function() {
                TweenLite.to(scene.getObjectById(toDel).scale, 0.4, {
                    x: this.scale,
                    y: this.scale,
                    z: this.scale,
                    ease:"Exponential.InOut",
                    onComplete:function(){
                        // console.log('tweeneeedddd');
                        // this.reverse()
                    }
                })
            }
        }
        tween2.go();

        document.getElementById("resultDIV").innerHTML = ("deleted: "+ toDel.object.id);
        editMode = false;
    }
}

```

**custom logic**  
MANAGING MULTIPLE GESTURES



# summary

*playful & exploratory > efficient & organized*

This app is meant to provide a playful way of interacting with the alarm clock through unique gestures. The purpose of the 3D environment is to encourage exploration in the user experience rather than efficiency.

For future implementations, customization as well as smoother animations can be added to make the experience more engaging.

