

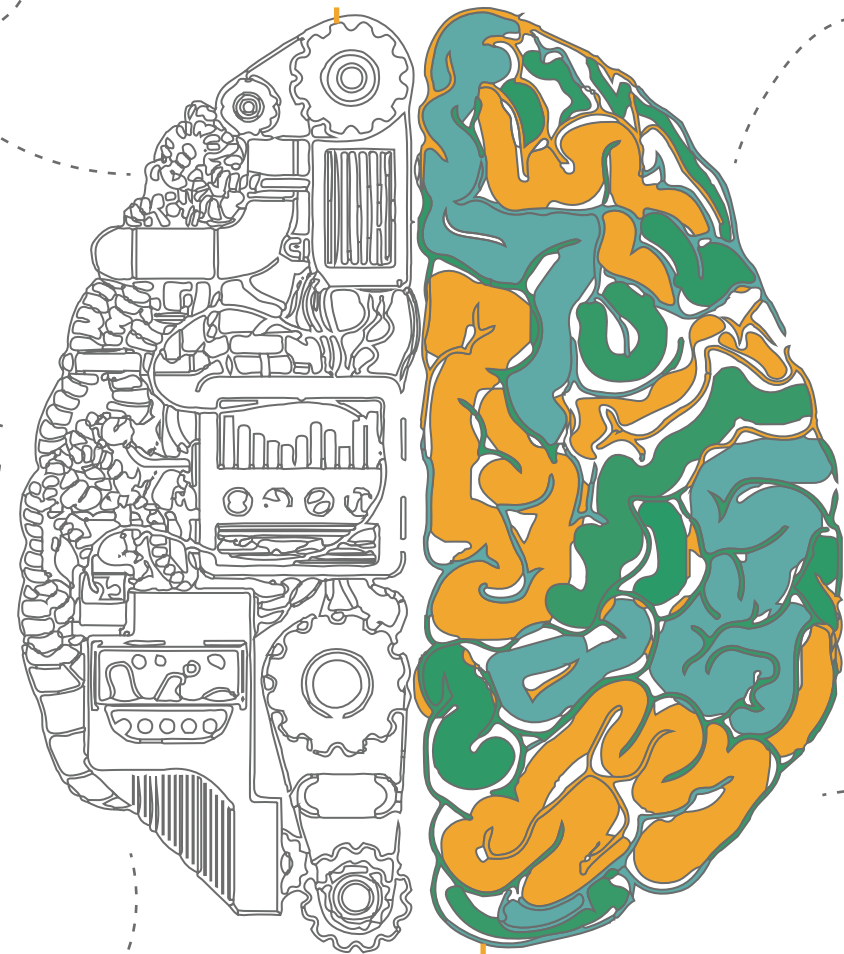
# IGNITE YOUR SENSES

Iconic spaces have the ability to excite the senses. It is possible to connect technology to the innate abilities of sensory that all human life possesses. We can celebrate the culture of innovation by revealing the relationship between advancement of technology and the human mind through design. By designing according to the likes of human experiences, spaces can be created for all.

Research in neuroaesthetics aims to understand biological and cognitive mechanisms that enable humans to have perceptual experiences in individually and culturally meaningful ways.

Aesthetic qualities in architecture have an impact on our mood, cognitive function, behavior and mental health.

Neurodiversity is a concept where neurological differences are recognized.



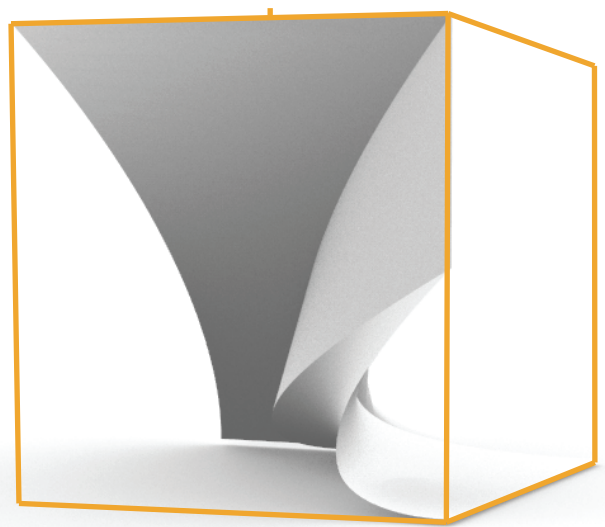
A study revealed that curvilinear building interiors are judged as more beautiful and pleasing than rectilinear spaces.<sup>1</sup>

Beauty ratings of curved rooms correlated with increased activation of anterior cingulate cortex. Which is the part of the brain's reward system.<sup>1</sup>

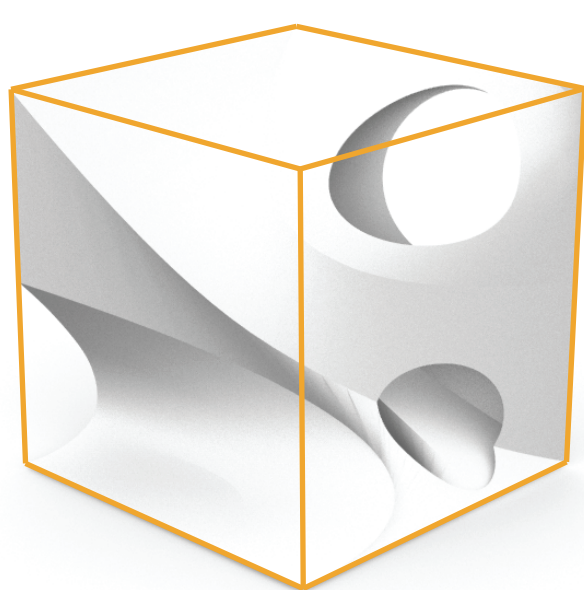
Humans perceive curves as less threatening than rectilinear shapes, and feel safe, thus creating a more inclusive environment.

This design reinforces inclusivity in which it is influenced through a combination of neuroaesthetics within architecture and neurodiversity.

Perception of curves and the preceding disciplines are the basis of the design, by appealing to the senses and creating an inclusive space through aesthetics.



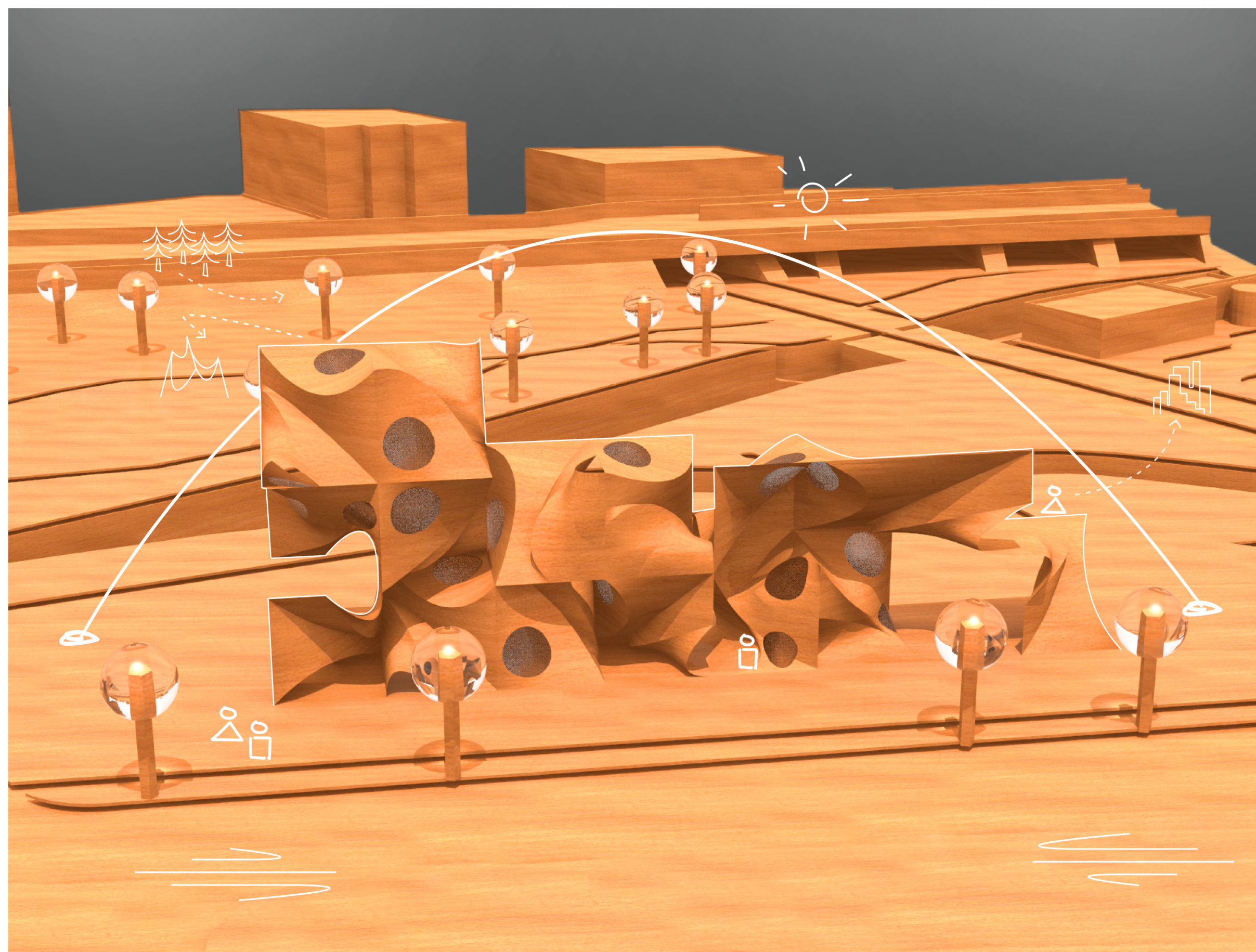
A single curvilinear object is constructed within a cubic boundary



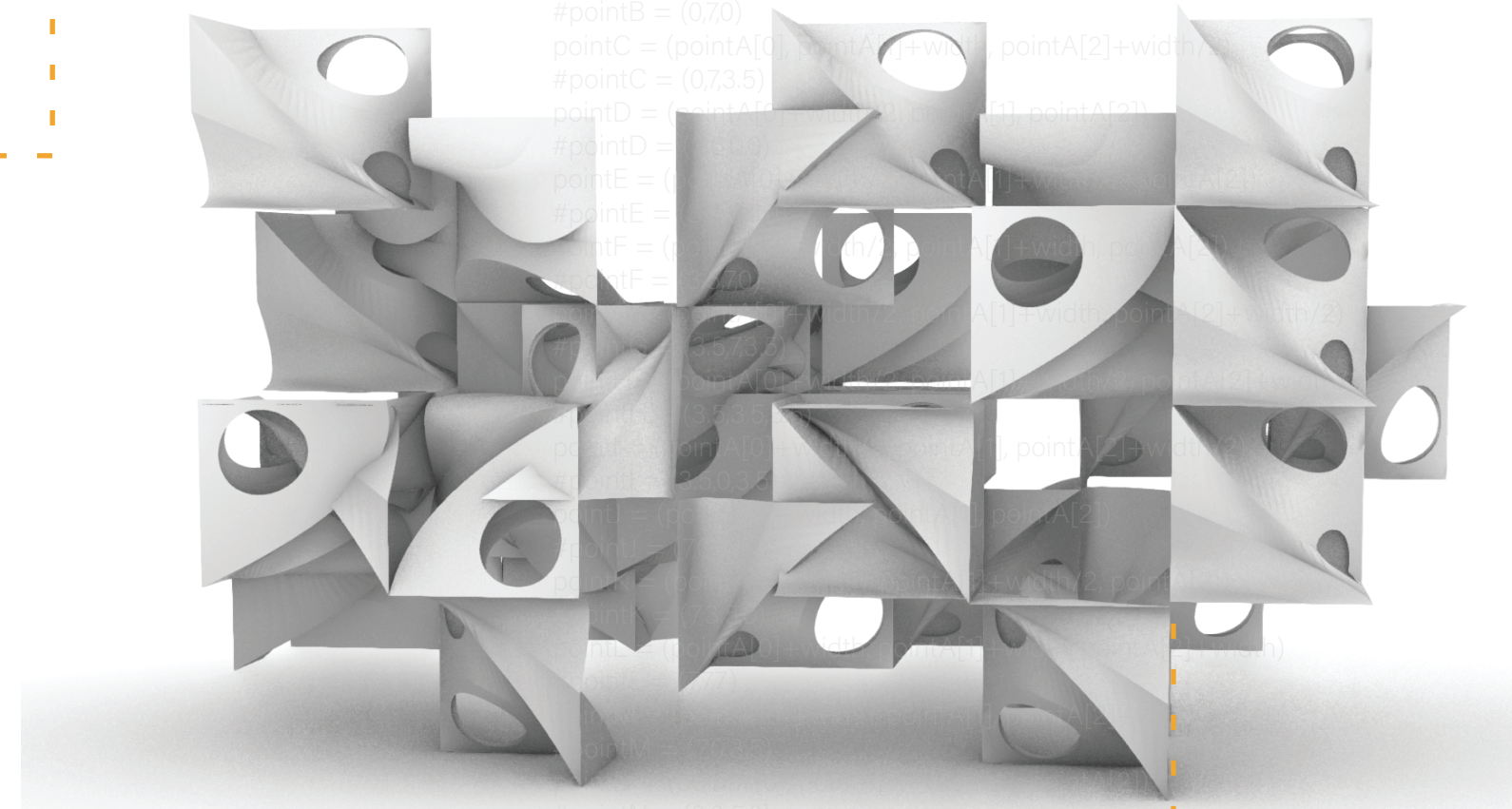
Singular module comprised of primarily 3-point curves



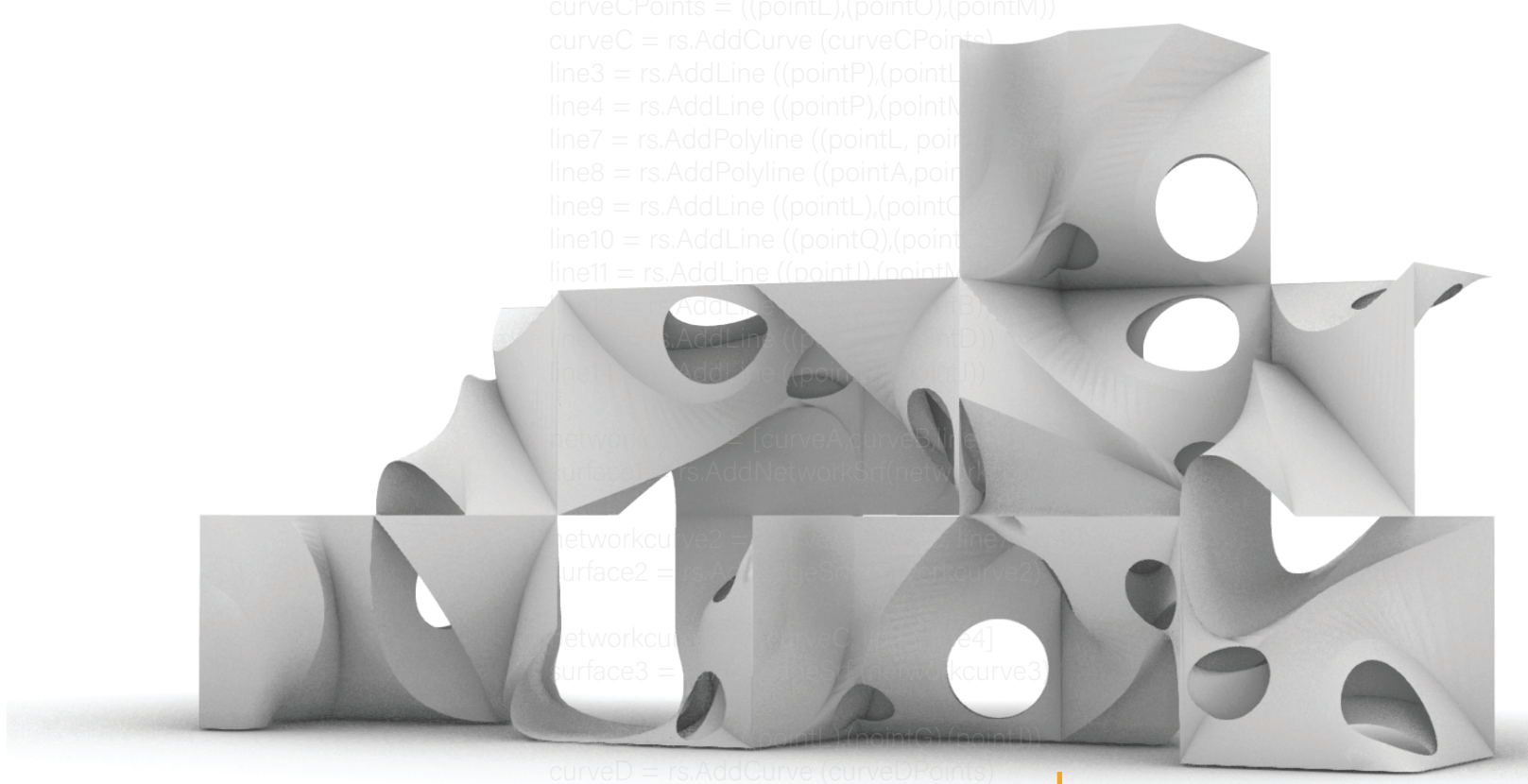
Single shape is asymmetrical for variation



Using a Python Script, the modules' aggregation allowed for spatial qualities and relations to neighboring modules to be investigated. The second study of the module aggregation reveals the object manipulated through random orthogonal rotation to reveal a mass with smoother flows of curves. Lastly, the aggregation of modules have curvilinear movement that is occasionally disrupted by linear geometry. This movement is evident in nature, that could mix with the landscape.



Complexity vs. Order



Manipulated Aggregation



Manipulated Aggregation

```

def __init__(self,midPoint,topPoint):
    self.midPoint = midPoint
    self.topPoint = topPoint
    self.coordinates = coordinates

    #self.coordinates = coordinates
    pointA = coordinates

    pointB = (pointA[0],pointA[1]+width,pointA[2])
    #pointB = (0,0,0)
    pointC = (pointA[0], pointA[1], pointA[2]+width)
    #pointC = (0,0,0)
    #pointD = (0,0,0)
    #pointE = (0,0,0)
    #pointF = (0,0,0)
    #pointG = (0,0,0)
    #pointH = (0,0,0)
    #pointI = (0,0,0)
    #pointJ = (0,0,0)
    #pointK = (0,0,0)
    #pointL = (0,0,0)
    #pointM = (0,0,0)
    #pointN = (0,0,0)
    #pointO = (0,0,0)
    #pointP = (0,0,0)
    #pointQ = (0,0,0)
    #pointR = (0,0,0)
    #pointS = (0,0,0)
    #pointT = (0,0,0)
    #pointU = (0,0,0)
    #pointV = (0,0,0)
    #pointW = (0,0,0)
    #pointX = (0,0,0)
    #pointY = (0,0,0)
    #pointZ = (0,0,0)

    curveAPoints = [(pointC),(pointI),(pointA)]
    curveA = rs.AddCurve (curveAPoints)
    curveBPoints = [(pointC),(pointB)]
    curveB = rs.AddCurve (curveBPoints)
    line1 = rs.AddLine ((pointN), (pointB))
    line2 = rs.AddLine ((pointN), (pointO))
    curveCPoints = [(pointI),(pointO),(pointM)]
    curveC = rs.AddCurve (curveCPoints)
    line3 = rs.AddLine ((pointP),(pointI))
    line4 = rs.AddLine ((pointP),(pointM))
    line7 = rs.AddPolyline ((pointL, pointM, pointN, pointO, pointP))
    line8 = rs.AddPolyline ((pointA, pointB, pointC, pointD, pointE, pointF, pointG, pointH, pointI, pointJ, pointK, pointL, pointM, pointN, pointO, pointP, pointQ, pointR, pointS, pointT, pointU, pointV, pointW, pointX, pointY, pointZ))
    line9 = rs.AddLine ((pointI),(pointM))
    line10 = rs.AddLine ((pointI),(pointM))
    line11 = rs.AddLine ((pointI),(pointM))

    networkCurve1 = [(line1), line3, line4, curveD]
    surface1 = rs.AddNetworkSrf(networkCurve1)

    networkCurve5 = [curveB, curveE, line2, line13]
    surface5 = rs.AddNetworkSrf(networkCurve5)

    networkCurve6 = [curveE, curveD, line4, line7]
    surface6 = rs.AddEdgesrf (networkCurve6)

    lines = [line1, line2, line3, line4, line7, line8, line9, line10, line11, line12, line13, line14]
    curves = [curveA, curveB, curveC, curveD, curveE]
    surfaces = [surface1, surface2, surface3, surface4, surface5, surface6]
    rs.DeleteObjects (lines)
    rs.DeleteObjects (curves)
    "Add holes in surface"
    finalObject = rs.JoinSurfaces ([surface1, surface2, surface3, surface4, surface5, surface6])
    rs.DeleteObjects (surfaces)
    self.finalObject = finalObject

    finalBox = rs.Box (finalObject)
    line (BBox(0), BBox(0))
    curveDomain(line)
    (1/2)*
    rs.EvaluateCurve(line, p)
    (pointI, pointM) = rs.AddPoint (lineCenter)

    Cheeses.append(0)
    Cheeses.append(1)
    Cheeses.append(k)
    Cheeses.append(j)
    Cheeses.append(i)
    BakedCheese.append(Cheeses)
    Cheeses = []
    rs.Redraw()

    relatedCheese = []
    X = rs.VectorCreate (0,0,0)(0,0,0)
    Y = rs.VectorCreate (0,0,0)(0,0,0)
    Z = rs.VectorCreate (0,0,0)(0,0,0)
    Planes = [X*Z]
    planrot = randomChoice(Planes)

    for j in BakedCheese:
        finalObject = finalObject
        rs.DeleteObject(0)(finalObject)
        (0) = 1
        rs.Redraw()

    for k in BakedCheese:
        rotateChoice = [0]
        ransrot = random (0,360) (deg)
        finalObject = rs.RotateObject (finalObject,k(0).centerPoint,ransrot, planrot)
        relatedCheese.append(relatedCheese)
        rs.Redraw()

    #Vertical Scale
    for i in BakedCheese:
        and ([2] == 0 and ([3] != 0)

    BakedCheese:
        and ([0] and ([3] == x) and ([4] == x)
        rs.MoveObject(0)(finalObject(0).midpoint,(1,2))
        rs.Redraw()

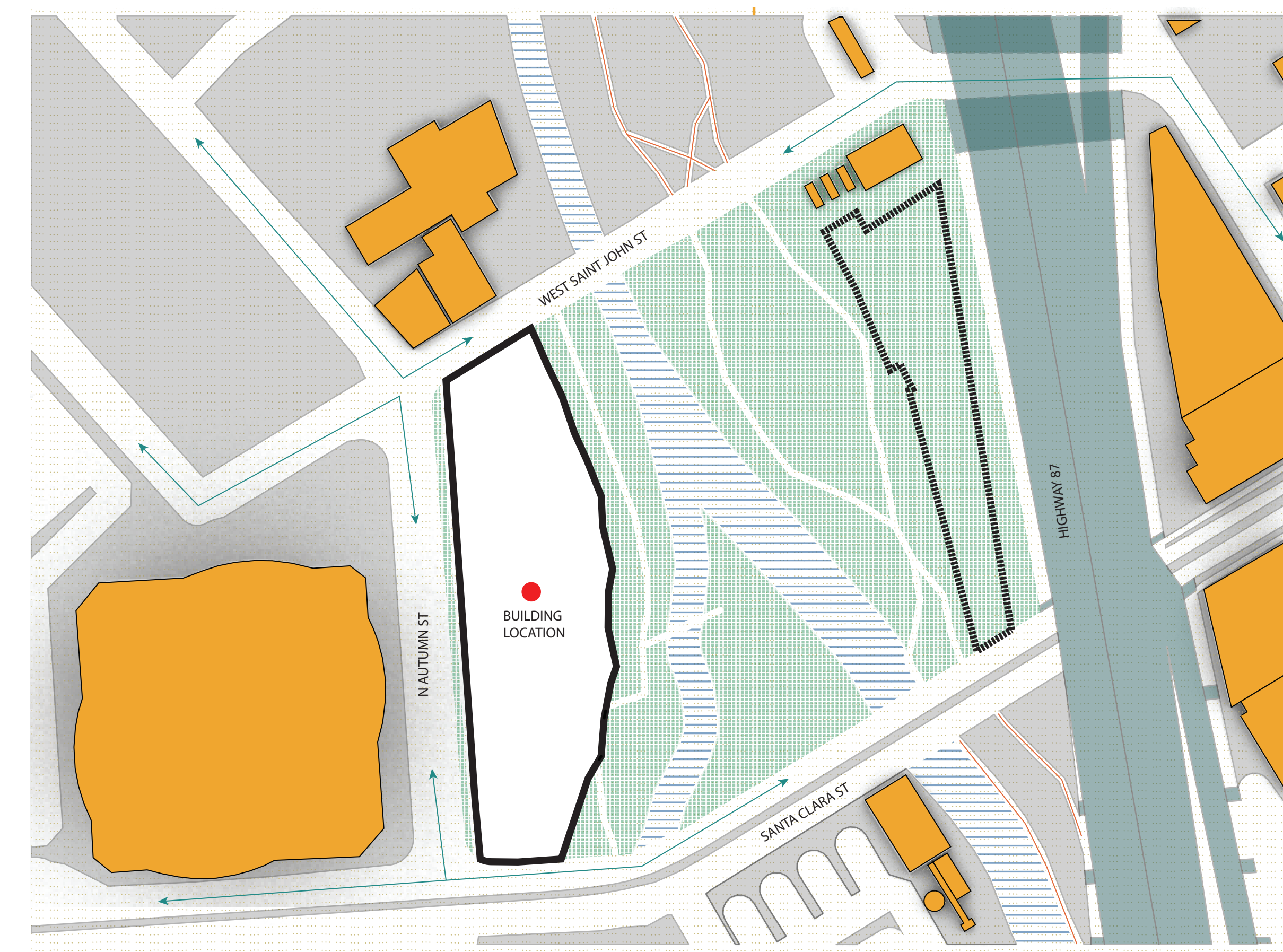
    #Horizontal Scale
    if ([1] == 1 and ([2])=0
    y = ([3]
    x = ([3]
    x = ([4]
    for i in BakedCheese:
        if ([2]=0 and ([2] == y-1 and ([3] == z and ([4] == x

```

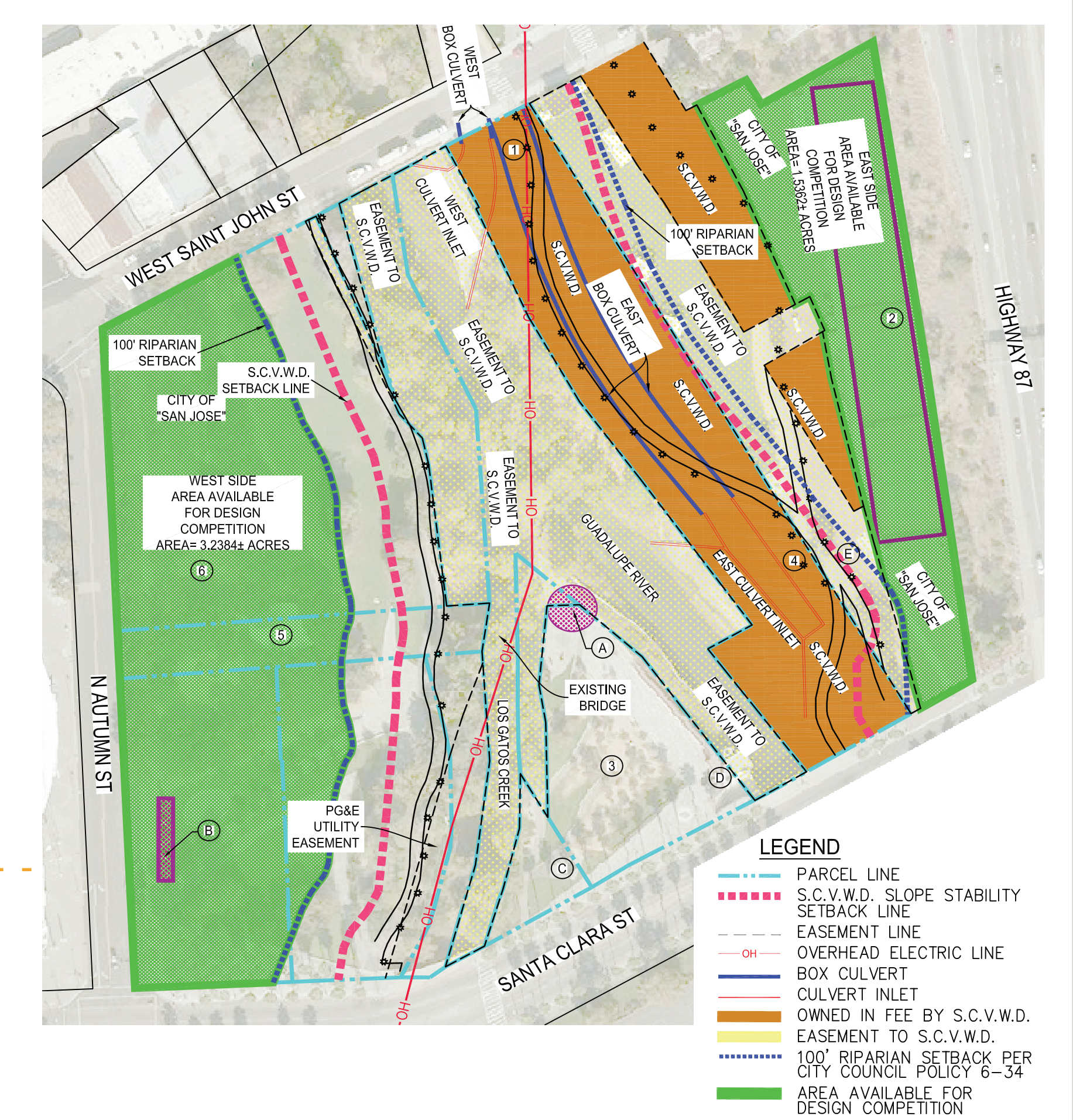


This space as an experience would welcome diverse communities throughout the world. The project's modularity allows sections to be created off-site. The site utilizes net zero site energy, creating renewable energy placed on the exterior of the modules. At night, the modular emit a low intense light, and the exterior material can be seen at a distance during the day. By emitting a low intensity light, the project remains environmentally sensitive to the surroundings.

Iconic spaces have the ability to excite the senses. It is possible to connect technology to the innate sensory abilities that all human life possesses. We can celebrate the culture of innovation by revealing the relationship between the advancement of technology and the human mind through design. By designing according to the likes of human experiences within the environment and highlighting the advancement of intelligence, spaces can be created for all.



The west side of the site is where the project is located. Locals and travelers will be able to walk through both sides of the site. The current east side of the site remains as a green space for the city and is uninhabited. The green spaces comprised on both sides will remain as service areas such as botanical gardens, event spaces and parks for the public.



- LEGEND**
- PARCEL LINE
  - S.C.V.W.D. SLOPE STABILITY SETBACK LINE
  - EASEMENT LINE
  - OH OVERHEAD ELECTRIC LINE
  - BOX CULVERT
  - CULVERT INLET
  - OWNED IN FEE BY S.C.V.W.D.
  - EASEMENT TO S.C.V.W.D.
  - 100' RIPARIAN SETBACK PER CITY COUNCIL POLICY 6-34
  - AREA AVAILABLE FOR DESIGN COMPETITION