



you desire. we design.





COMPANY OVERVIEW

Founded in Turin in **2013**

Engineering and Design consultancy Company

140 people across 2 locations: **Moncalieri (TO), Formigine (MO)**
India and Brazil (work in progress)

Turnover **8.5 mil €** in 2024

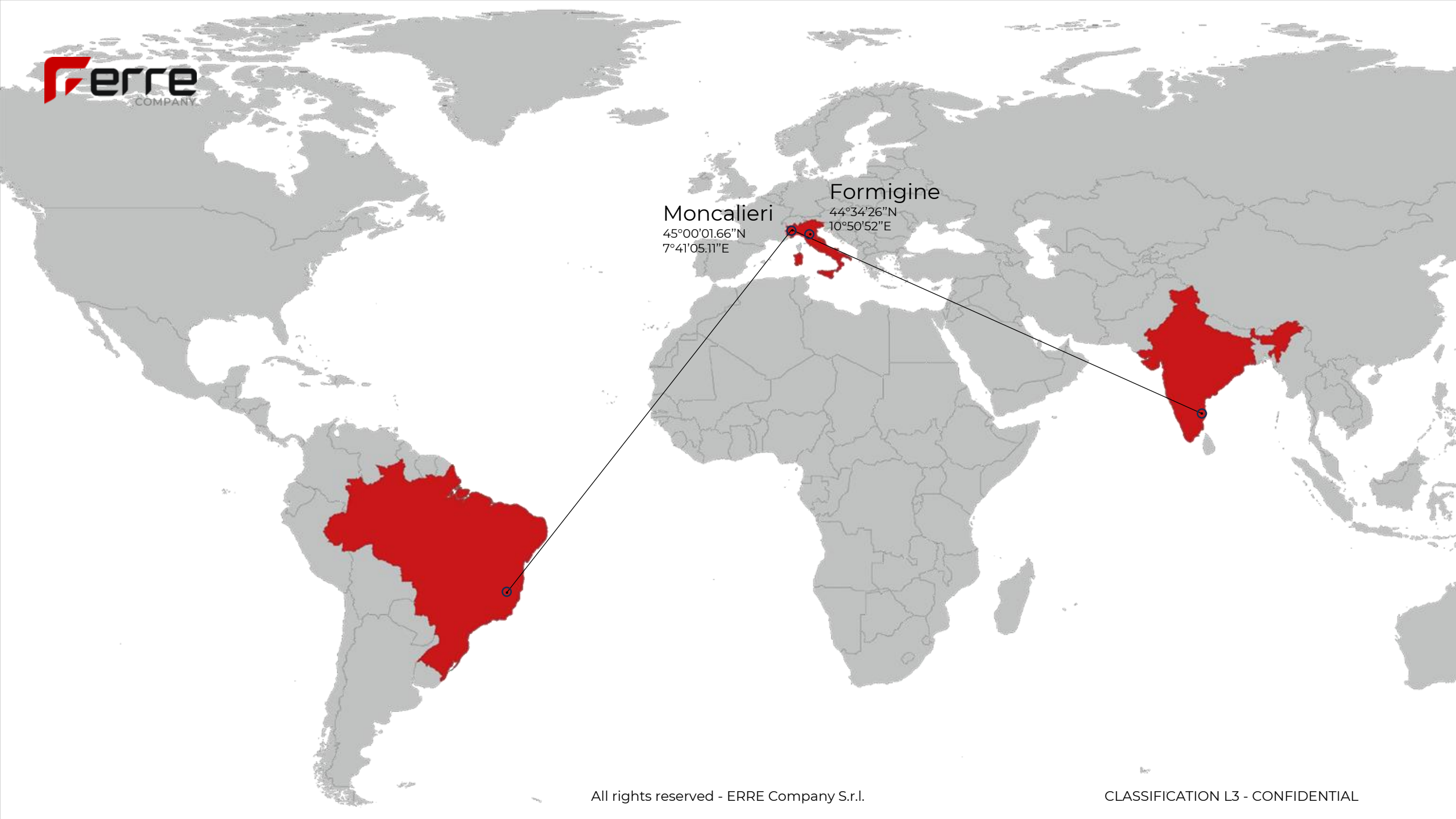
Associated to:



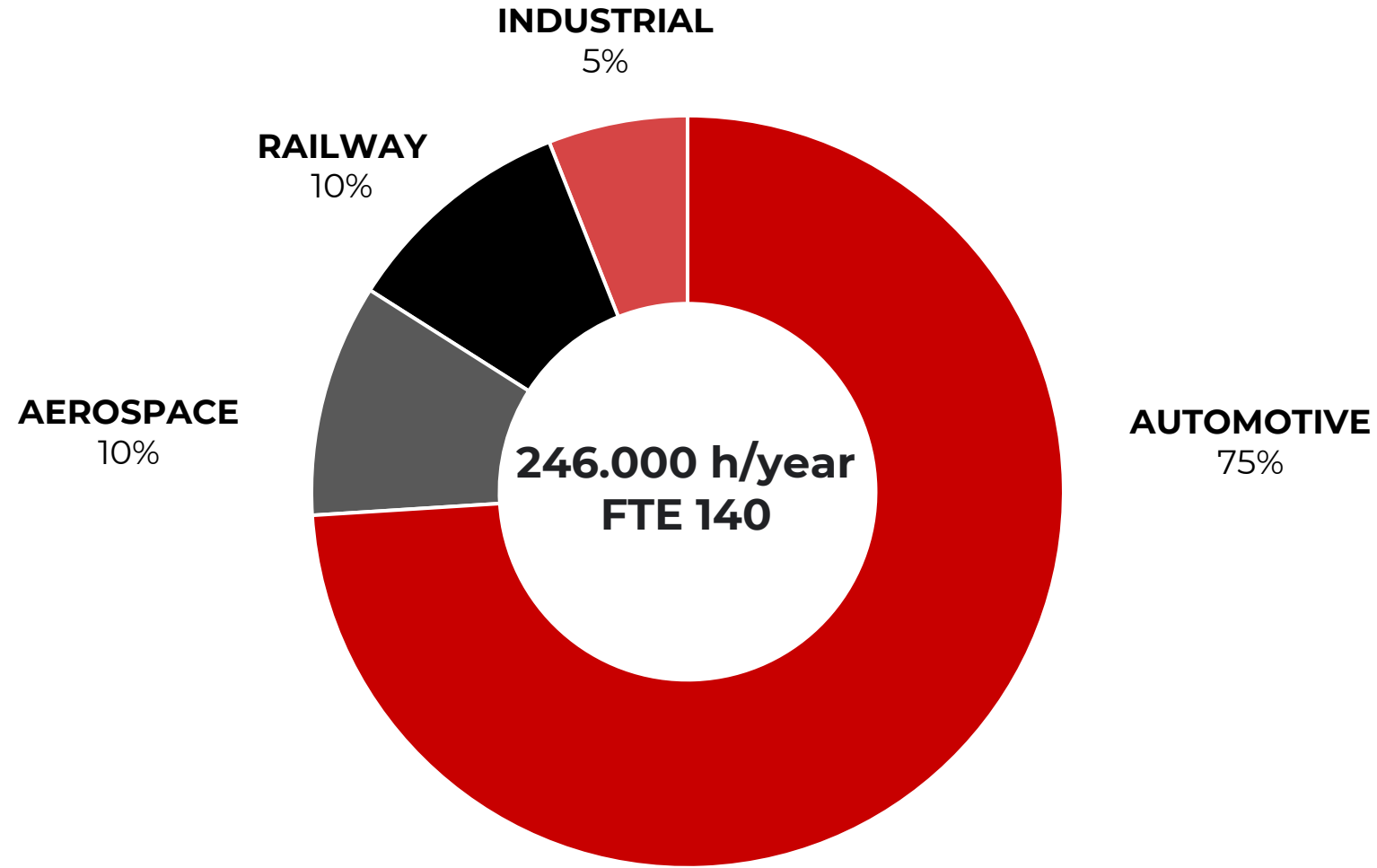
Certifications:



ISO 9001
EN 9100



BUSINESS AREAS

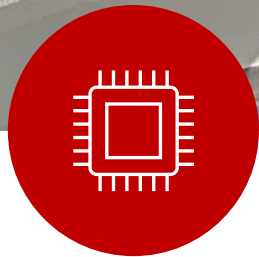




Engineering



Design Studio



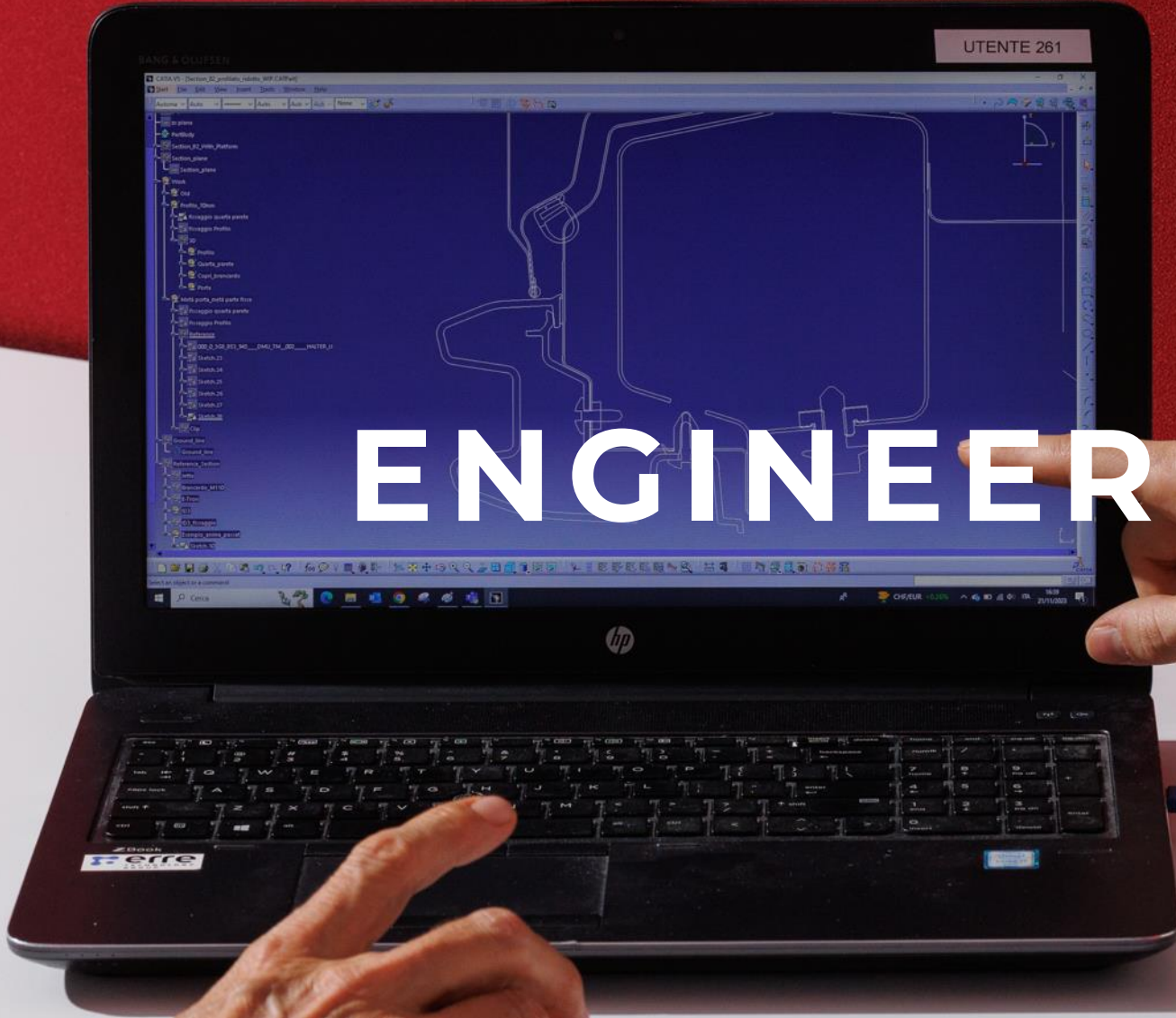
EES



ICT

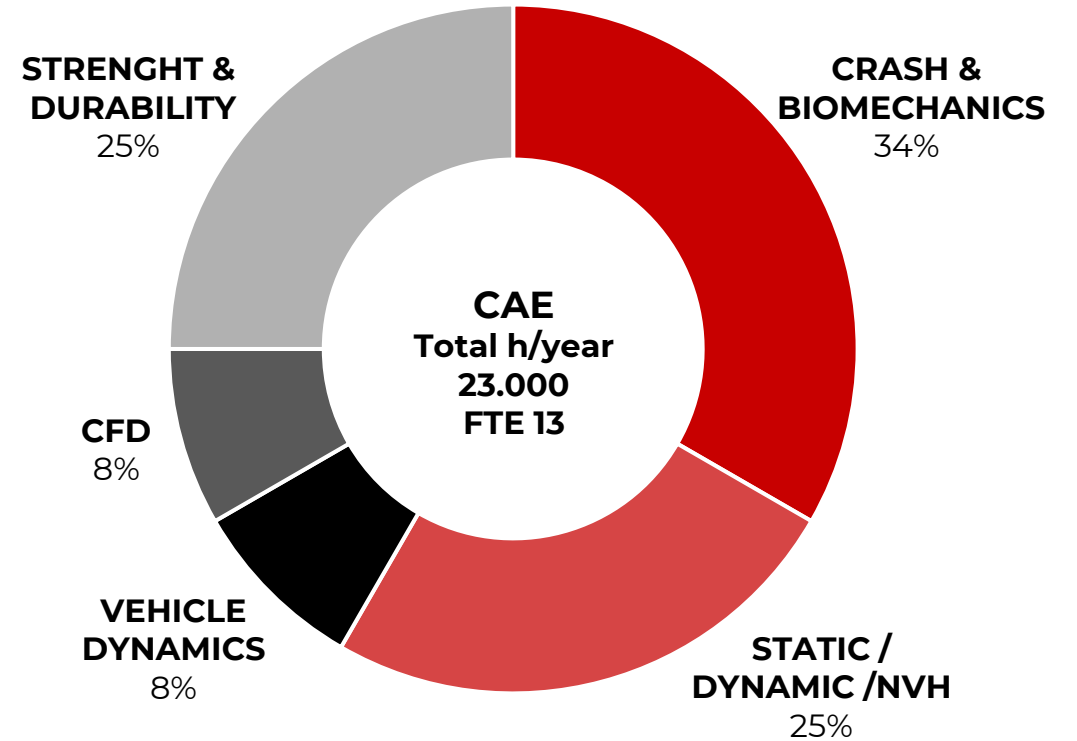
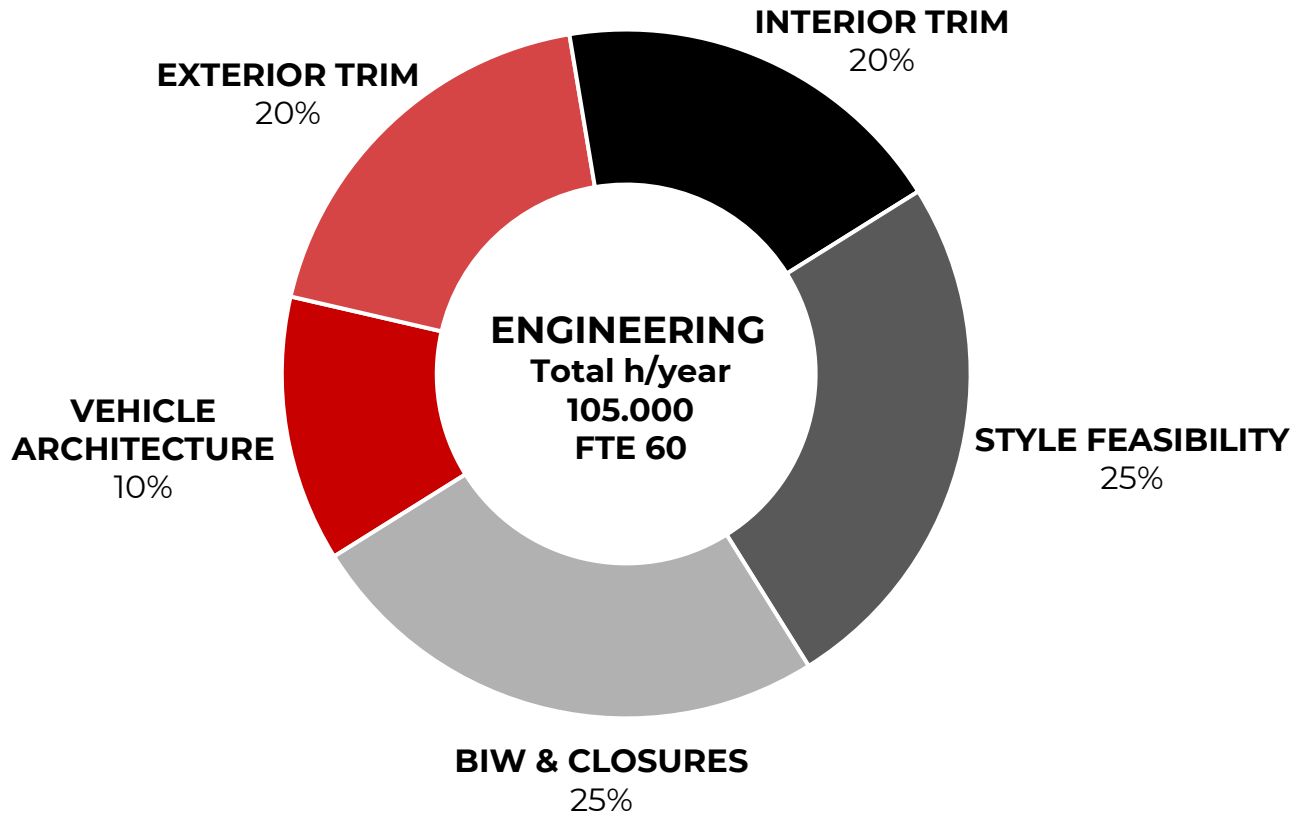


Healthcare



ENGINEERING

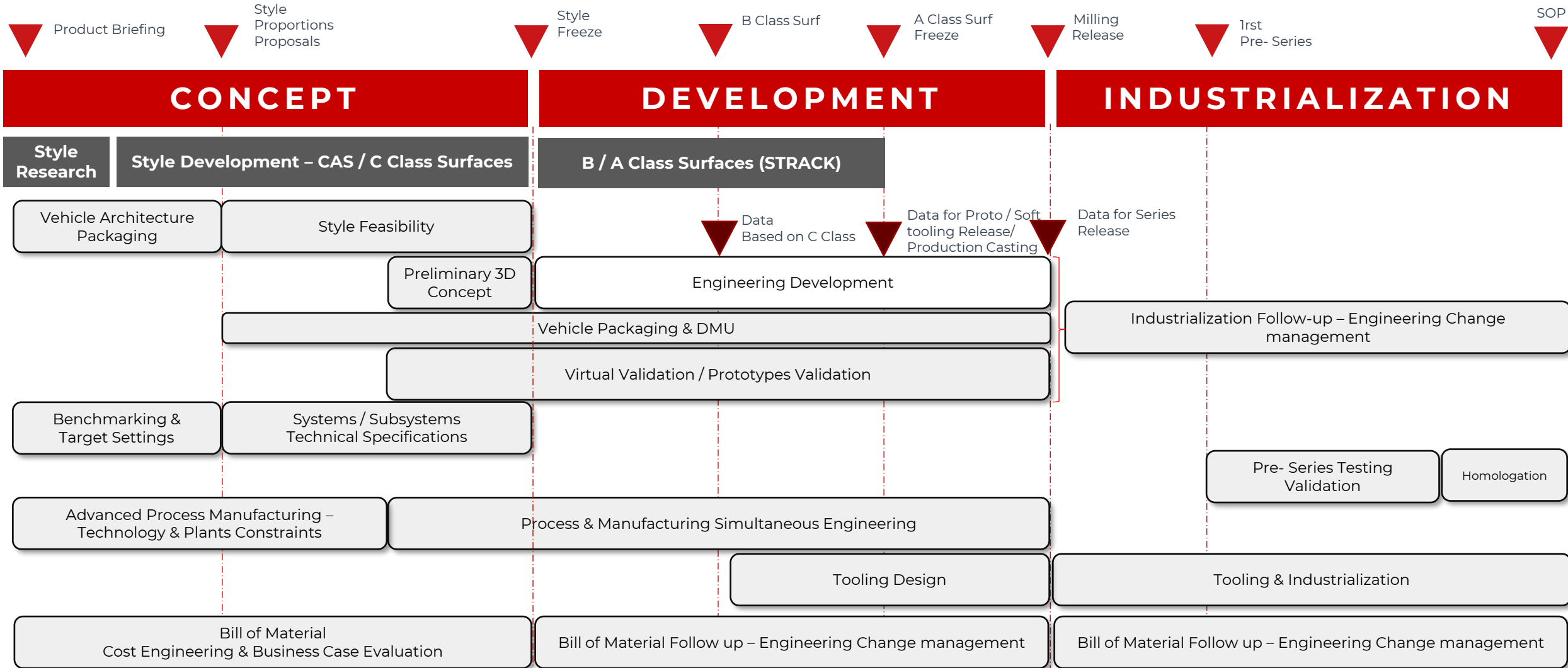
ENGINEERING COMPONENT/SYSTEM CAPACITY



PRODUCT DEVELOPMENT PROCESS DESIGN



ERRE covers the whole Product Development Process from the early stage of Concept Ideation till the Starting of Production





Product Briefing

Style Proportions Proposals

Style Freeze

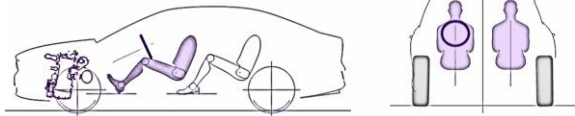
B Class Surf

A Class Surf Freeze

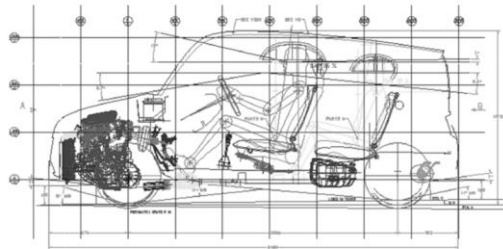
Milling Release

1st Pre-Series

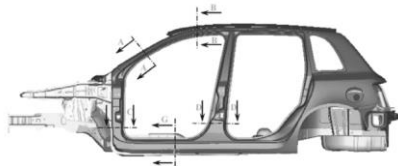
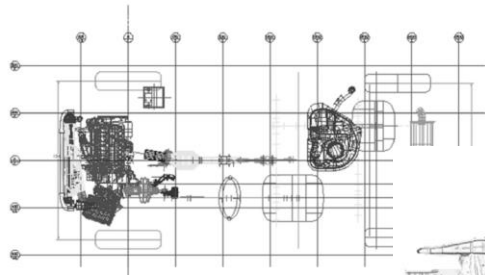
CONCEPT



Packaging Ideation / dimensions / homolog set up



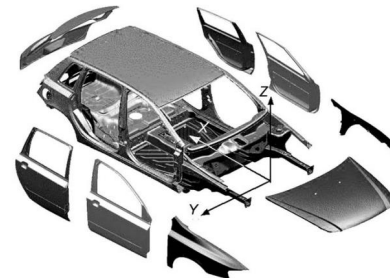
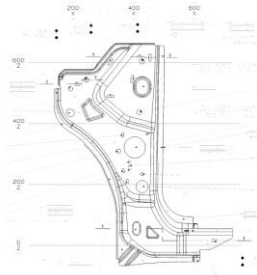
Vehicle Occupant Packaging & Hard Points



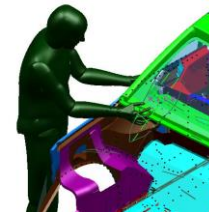
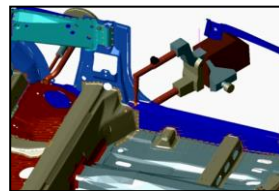
Feasibility Studies

DEVELOPMENT

Product & Process Simultaneous Engineering



Product



Process

Virtual & Physical Validation



INDUSTRIALIZATION

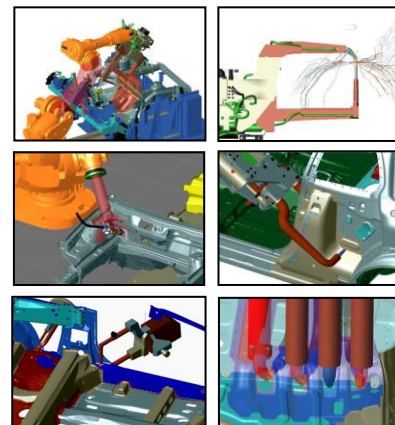
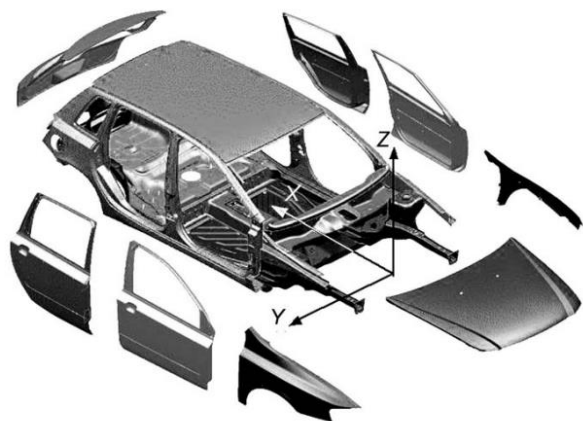


Product & Process Engineering Change Management

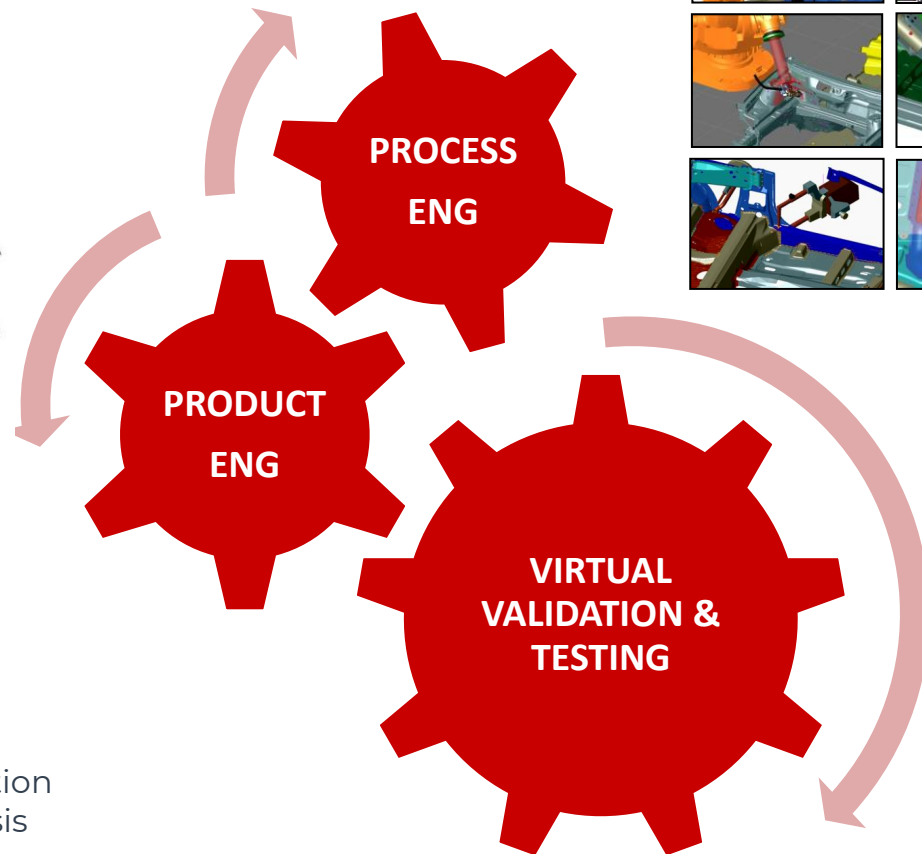
Final Product and Process Validation

Production Quality

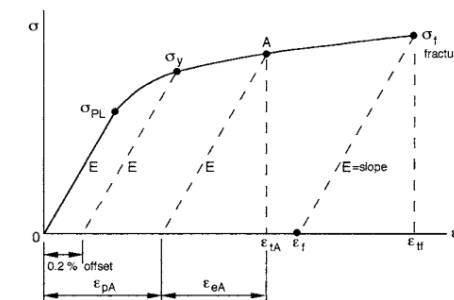
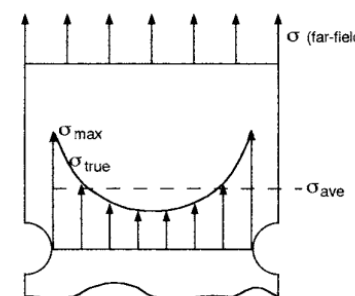
SIMULTANEOUS ENGINEERING



- Formability Analysis
- Body Assembly
- Trims & Systems Final Assembly
- Painting
- Logistics



- BIW & Closures
- Ext/Int trim,
- E/E Wiring Harness
- E/E HW & SW Dev
- Vehicle Packaging
- PWT & Driveline Installation
- Platform & Rolling Chassis
- Bill Of Materials
- DMU
- Tolerance Chain Analysis



- Crash Analyses (Structural, Sensors, Biomechanics)
- Static, Dynamic & NVH analyses
- Fatigue/Durability analysis
- Vehicle dynamics (Multibody)
- CFD & Multi-physics analyses
- Robust Design and Optimization

TESTING LAB

Ferre
COMPANY

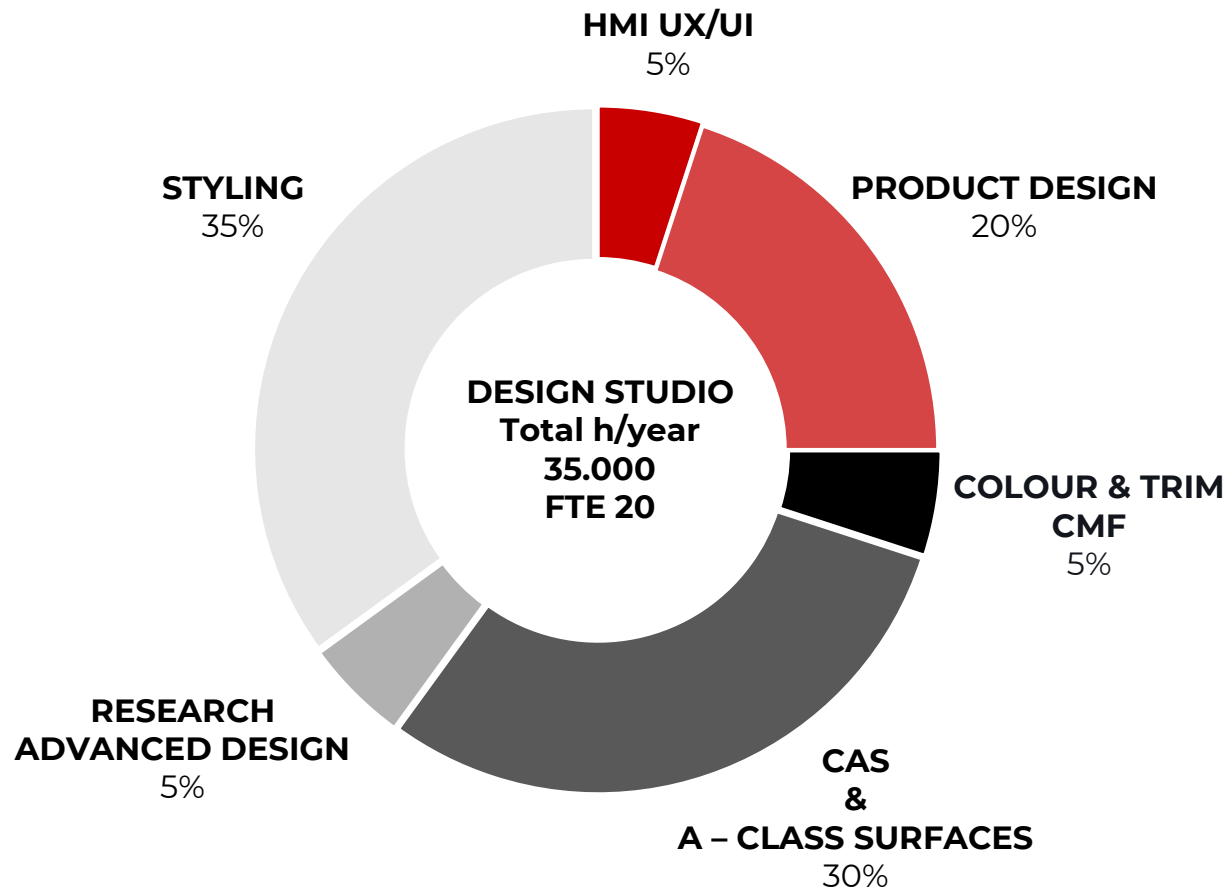




DESIGN STUDIO



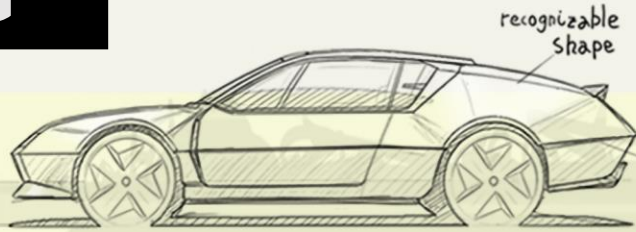
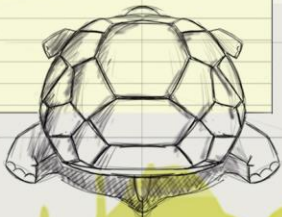
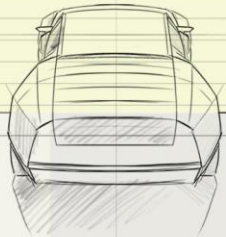
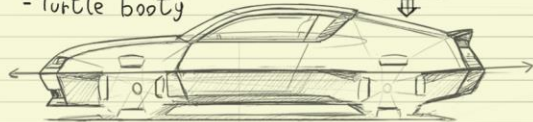
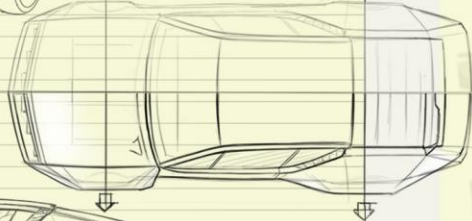
DESIGN STUDIO CAPACITY



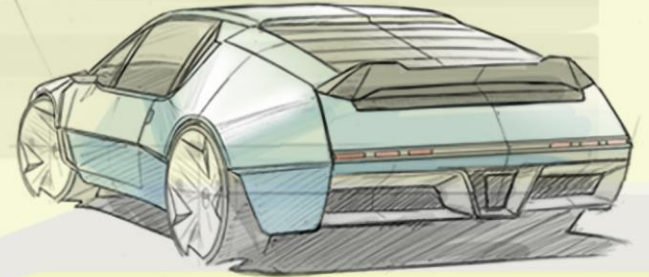
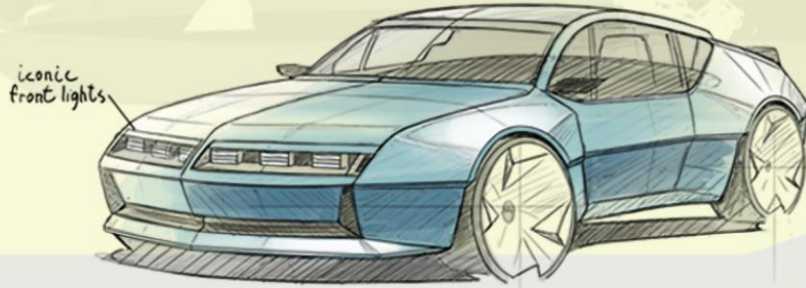
STYLING

Concept.

- Two shells
- Wide body
- Division line
- Turtle booty

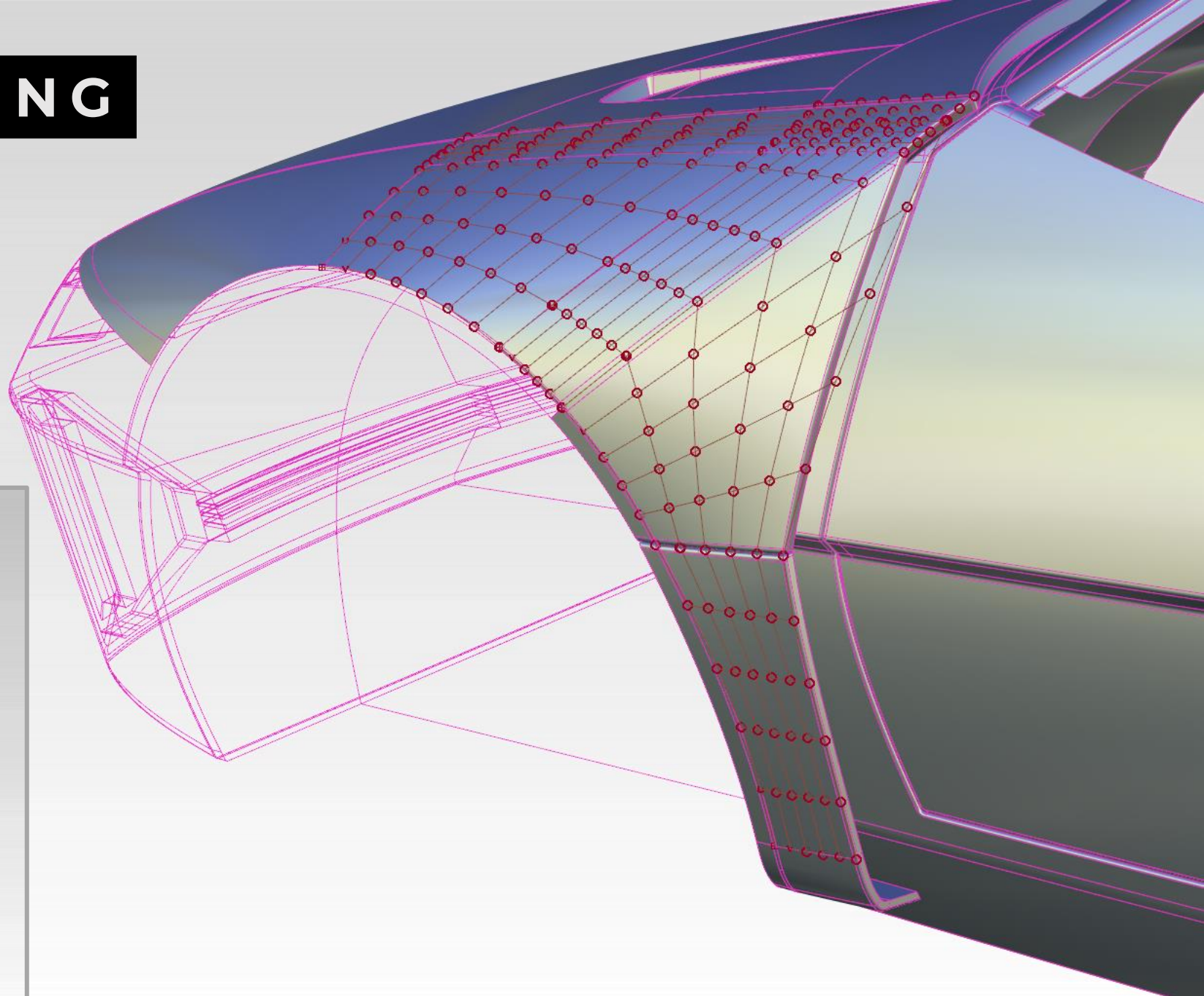
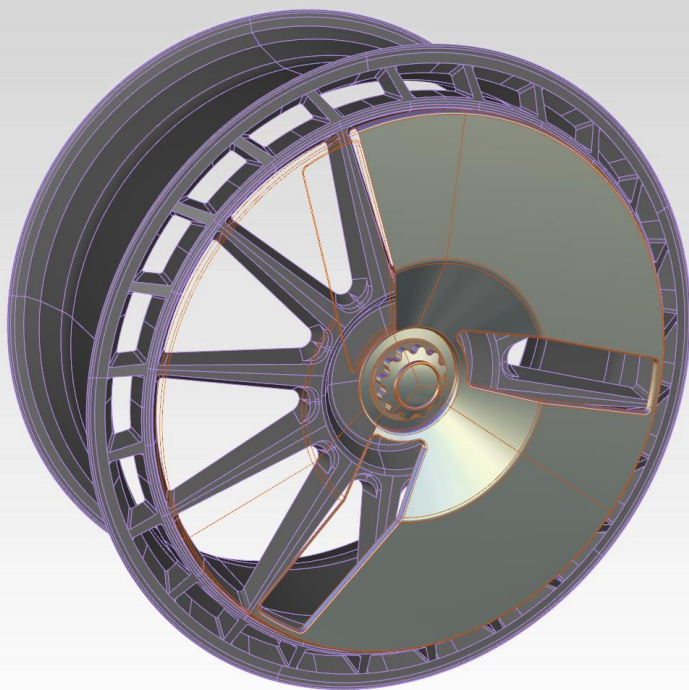


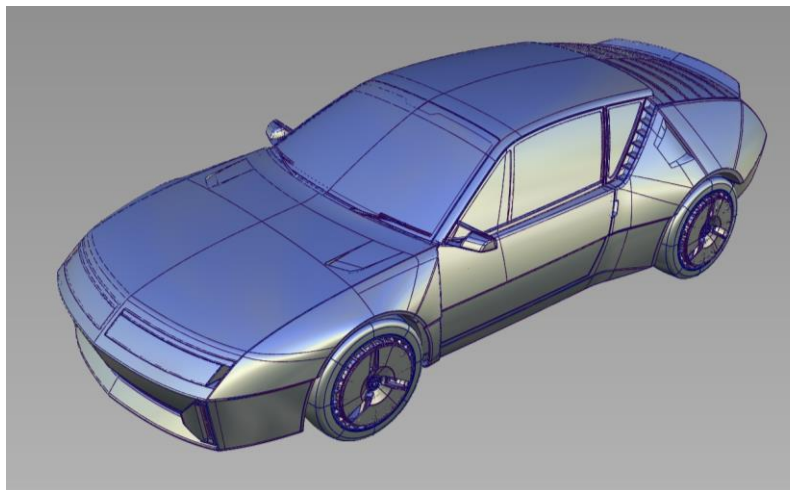
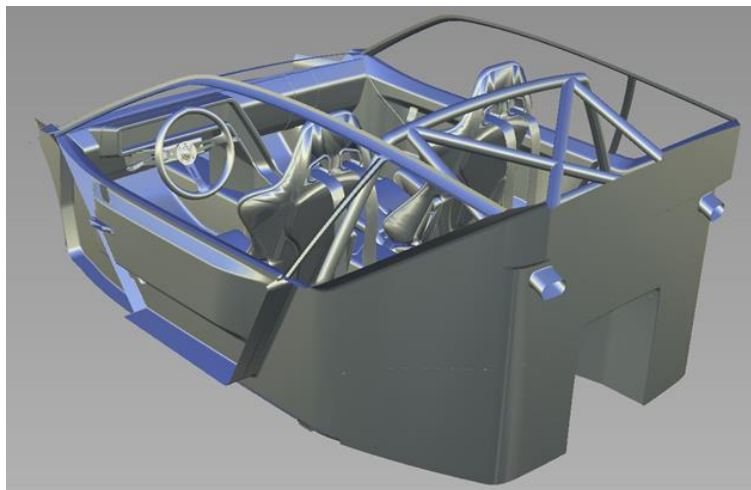
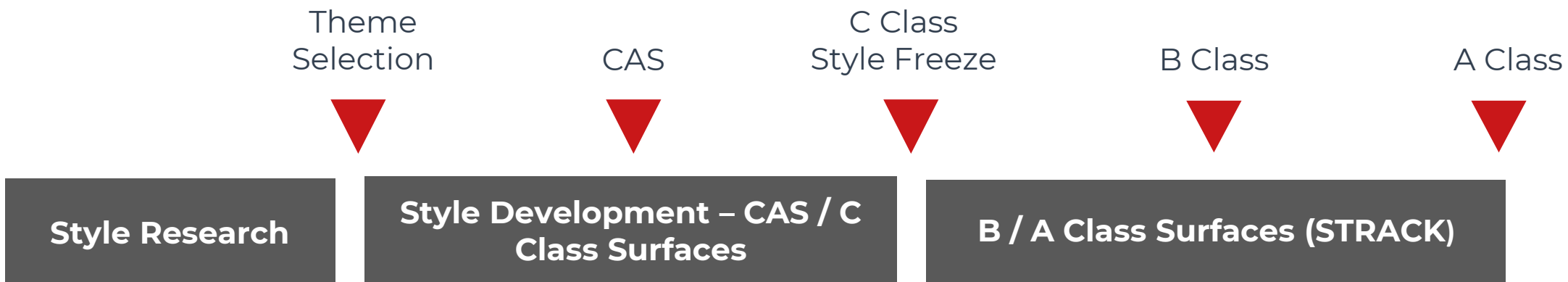
First sketches.



-Design proposal.

DIGITAL MODELING





Style Research

- Adobe Photoshop
- Blender
- Autodesk Alias SUB-D
- After Effect / Figma (HMI)
- Autodesk Vred
- KeyShot 3D
- Illustrator

Surfaces Modeling

- Autodesk Alias
- ICEM Surf

CMF





AWARD 2025



PRODUCTION CARS
FERRARI
12CILINDRI

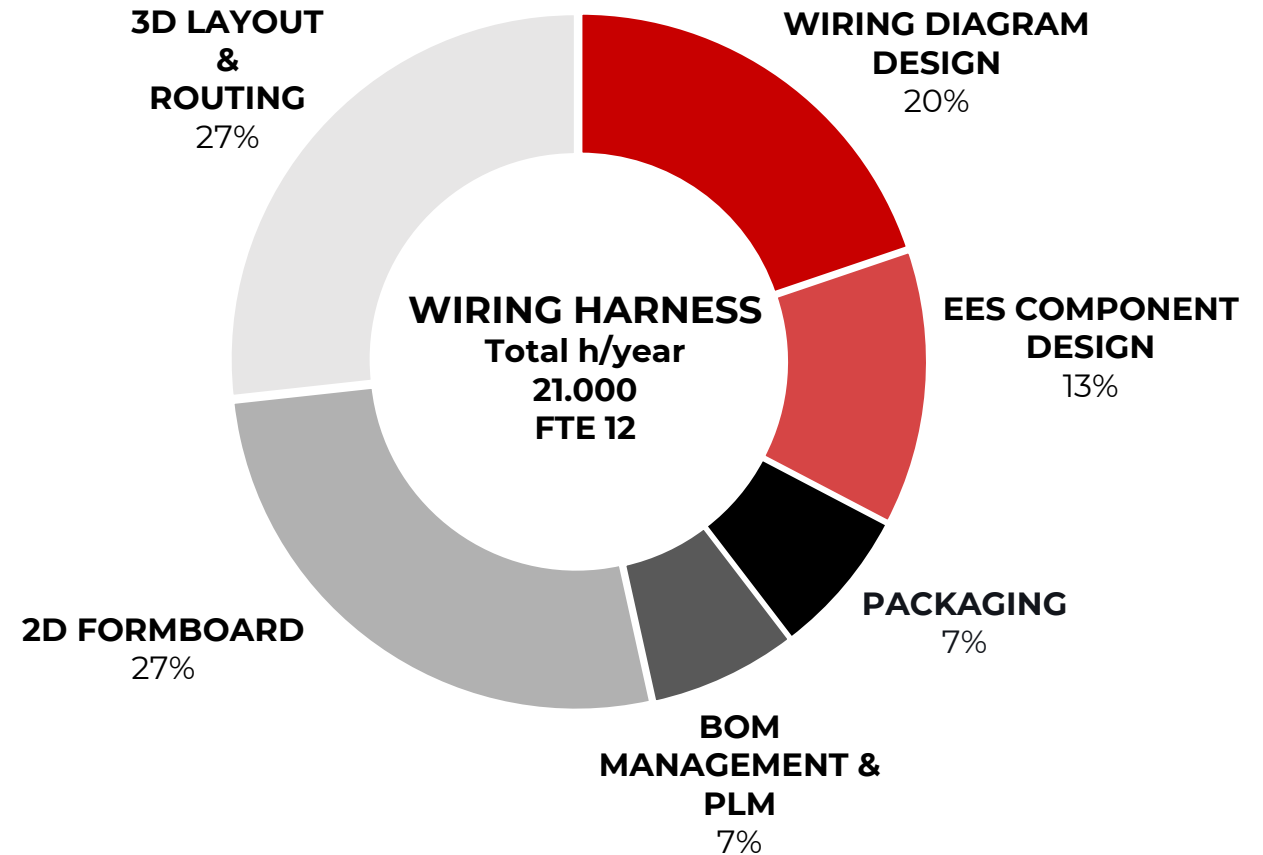
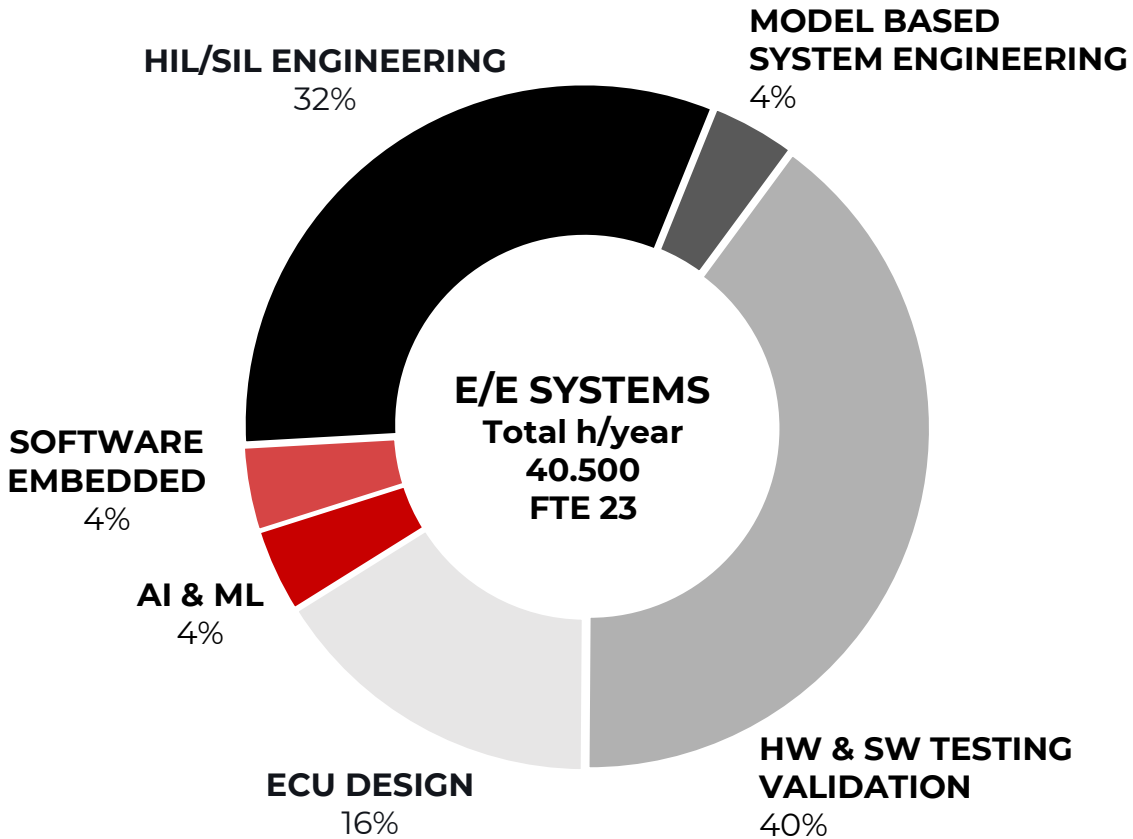


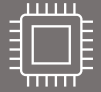
«TRACCIA»

EES



E/E SYSTEMS COMPONENTS





E/E HW-SW SYSTEM ENGINEERING & DESIGN

- EE embedded SW and HMI system development
- model based system engineering & architecture design
- artificial intelligence (ai), machine learning, big data analysis
- EE system requirement definition and functional homologation



E/E PRODUCT & PROJECT DESIGN

- wiring harness 3D/2D design
- wiring diagram & schematic design
- component design & packaging (plastic-metal type)
- ecu design and HW layout



E/E WORKSHOP & TESTING LABORATORY

- EE hardware production and commissioning
- wiring harness prototyping production
- systems validation (physical mock up, on-vehicle instrumentation, bench-testing, calibration, benchmarking, homologation)
- EEcontrol unit testing and system validation (hil-sil)

EES LAB

Ferre
COMPANY





PRODUCTION PHASES

- Cut and crimping
- Twisting
- Assembly
- Soldering (Splicing)
- Testing, CQC and Packaging





ICT

```

File Edit Selection View Go Run Terminal Help
Automation1.py X
1 import cv2
2 import mediapipe as mp
3 import numpy as mp
4 mp_drawing = mp.solutions.drawing_utils
5 mp_hands = mp.solutions.hands
6 mp_pose = mp.solutions.pose
7 cap = cv2.VideoCapture(0)
8 hands = mp_hands.Hands()
9 pose = mp_pose.Pose(min_detection_confidence=0.5, min_tracking_confidence=0.5)
10 def draw_avatar(image, landmarks, height, width):
11     points = []
12     for idx in [11, 12, 13, 14, 15, 16, 23, 24, 25, 26, 27, 28]: # shoulders, elbows, wrists, hips, knees
13         if landmarks[idx].visibility > 0.5:
14             points[idx] = (int(landmarks[idx].x * width), int(landmarks[idx].y * height))
15     center_x, center_y = width // 2, height // 4
16     avatar_points = {
17         'head': (center_x, center_y),
18         'left_shoulder': (center_x - 50, center_y + 50),
19         'right_shoulder': (center_x + 50, center_y + 50),
20         'left_elbow': (center_x - 75, center_y + 100),
21         'right_elbow': (center_x + 75, center_y + 100),
22         'left_wrist': (center_x - 100, center_y + 150),
23         'right_wrist': (center_x + 100, center_y + 150),
24         'left_hip': (center_x - 50, center_y + 200),
25         'right_hip': (center_x + 50, center_y + 200),
26         'left_knee': (center_x - 50, center_y + 300),
27         'right_knee': (center_x + 50, center_y + 300),
28         'left_ankle': (center_x - 50, center_y + 400),
29         'right_ankle': (center_x + 50, center_y + 400),
30     }
31     cv2.line(image, avatar_points['head'], avatar_points['left_shoulder'], (255, 0, 0), 3)
32     cv2.line(image, avatar_points['head'], avatar_points['right_shoulder'], (255, 0, 0), 3)
33     cv2.line(image, avatar_points['left_shoulder'], avatar_points['left_elbow'], (255, 0, 0), 3)
34     cv2.line(image, avatar_points['right_shoulder'], avatar_points['right_elbow'], (255, 0, 0), 3)
35     cv2.line(image, avatar_points['left_elbow'], avatar_points['left_wrist'], (255, 0, 0), 3)
36     cv2.line(image, avatar_points['right_elbow'], avatar_points['right_wrist'], (255, 0, 0), 3)
37     cv2.line(image, avatar_points['left_wrist'], avatar_points['left_hip'], (255, 0, 0), 3)
38     cv2.line(image, avatar_points['right_wrist'], avatar_points['right_hip'], (255, 0, 0), 3)
39     cv2.line(image, avatar_points['left_hip'], avatar_points['left_knee'], (255, 0, 0), 3)
40     cv2.line(image, avatar_points['right_hip'], avatar_points['right_knee'], (255, 0, 0), 3)
41     cv2.line(image, avatar_points['left_knee'], avatar_points['left_ankle'], (255, 0, 0), 3)
42     cv2.line(image, avatar_points['right_knee'], avatar_points['right_ankle'], (255, 0, 0), 3)
43     cv2.line(image, avatar_points['left_ankle'], avatar_points['left_hip'], (255, 0, 0), 3)
44     cv2.line(image, avatar_points['right_ankle'], avatar_points['right_hip'], (255, 0, 0), 3)
45     return image
46
47 while True:
48     success, image = cap.read()
49     if not success:
50         break
51     pose_landmarks = pose.solve(image)
52     if pose_landmarks is None:
53         continue
54     image = draw_avatar(image, pose_landmarks, image.shape[0], image.shape[1])
55     cv2.imshow('Avatar', image)
56     if cv2.waitKey(1) & 0xFF == ord('q'):
57         break
58     cv2.destroyAllWindows()
59
60 # Package list
61 packages: {
62   'dependencies': {
63     'angular/animations': '^15.2.10',
64     'angular/common': '^15.2.10',
65     'angular/core': '^15.2.10',
66     'angular/forms': '^15.2.10',
67     'angular/router': '^15.2.10',
68     'angular/service-worker': '^15.2.10',
69     'angular/ssr': '^15.2.10',
70     '@angular-devkit/build-angular': '^15.2.10',
71     '@angular-devkit/schematics': '^15.2.10',
72     '@angular/cli': '^15.2.10',
73     '@angular/compiler': '^15.2.10',
74     '@angular/compiler-cli': '^15.2.10',
75     '@angular/localize': '^15.2.10',
76     '@angular/platform-browser': '^15.2.10',
77     '@angular/platform-browser-dynamic': '^15.2.10',
78     '@angular/platform-server': '^15.2.10',
79     '@angular/pwa': '^15.2.10',
80     '@angular/ssr': '^15.2.10',
81     '@ngtools/webpack': '^15.2.10',
82     '@nguniversal/builders': '^15.2.10',
83     '@nguniversal/express-engine': '^15.2.10',
84     '@nguniversal/module-optimizer': '^15.2.10',
85     '@nguniversal/postcss-plugin': '^15.2.10',
86     '@nguniversal/router': '^15.2.10',
87     '@nguniversal/webpack-plugin': '^15.2.10',
88     '@nguniversal/webpack-plugin-express': '^15.2.10',
89     '@nguniversal/webpack-plugin-express-ssl': '^15.2.10',
90     '@nguniversal/webpack-plugin-express-ssl-https': '^15.2.10',
91     '@nguniversal/webpack-plugin-express-ssl-https-https': '^15.2.10',
92     '@nguniversal/webpack-plugin-express-ssl-https-https-https': '^15.2.10',
93     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https': '^15.2.10',
94     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https': '^15.2.10',
95     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https-https': '^15.2.10',
96     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https-https-https': '^15.2.10',
97     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https-https-https-https': '^15.2.10',
98     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https-https-https-https-https': '^15.2.10',
99     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https-https-https-https-https-https': '^15.2.10',
100    }
101   }
102 }

```

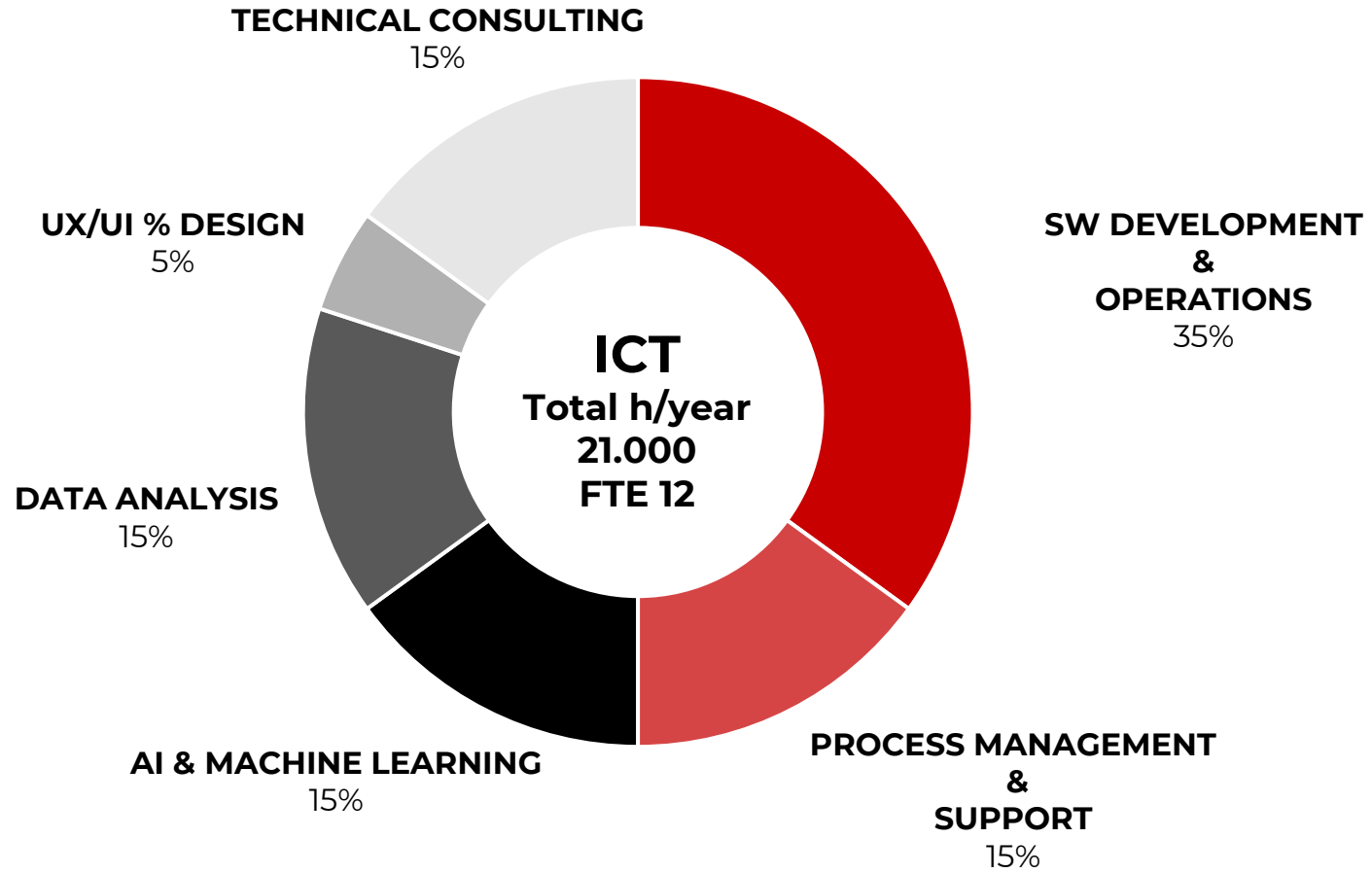
```

File Edit Selection View Go Run Terminal Help
Automation1.py X
1 import cv2
2 import mediapipe as mp
3 import numpy as mp
4 mp_drawing = mp.solutions.drawing_utils
5 mp_hands = mp.solutions.hands
6 mp_pose = mp.solutions.pose
7 cap = cv2.VideoCapture(0)
8 hands = mp_hands.Hands()
9 pose = mp_pose.Pose(min_detection_confidence=0.5, min_tracking_confidence=0.5)
10 def draw_avatar(image, landmarks, height, width):
11     points = []
12     for idx in [11, 12, 13, 14, 15, 16, 23, 24, 25, 26, 27, 28]: # shoulders, elbows, wrists, hips, knees
13         if landmarks[idx].visibility > 0.5:
14             points[idx] = (int(landmarks[idx].x * width), int(landmarks[idx].y * height))
15     center_x, center_y = width // 2, height // 4
16     avatar_points = {
17         'head': (center_x, center_y),
18         'left_shoulder': (center_x - 50, center_y + 50),
19         'right_shoulder': (center_x + 50, center_y + 50),
20         'left_elbow': (center_x - 75, center_y + 100),
21         'right_elbow': (center_x + 75, center_y + 100),
22         'left_wrist': (center_x - 100, center_y + 150),
23         'right_wrist': (center_x + 100, center_y + 150),
24         'left_hip': (center_x - 50, center_y + 200),
25         'right_hip': (center_x + 50, center_y + 200),
26         'left_knee': (center_x - 50, center_y + 300),
27         'right_knee': (center_x + 50, center_y + 300),
28         'left_ankle': (center_x - 50, center_y + 400),
29         'right_ankle': (center_x + 50, center_y + 400),
30     }
31     cv2.line(image, avatar_points['head'], avatar_points['left_shoulder'], (255, 0, 0), 3)
32     cv2.line(image, avatar_points['head'], avatar_points['right_shoulder'], (255, 0, 0), 3)
33     cv2.line(image, avatar_points['left_shoulder'], avatar_points['left_elbow'], (255, 0, 0), 3)
34     cv2.line(image, avatar_points['right_shoulder'], avatar_points['right_elbow'], (255, 0, 0), 3)
35     cv2.line(image, avatar_points['left_elbow'], avatar_points['left_wrist'], (255, 0, 0), 3)
36     cv2.line(image, avatar_points['right_elbow'], avatar_points['right_wrist'], (255, 0, 0), 3)
37     cv2.line(image, avatar_points['left_wrist'], avatar_points['left_hip'], (255, 0, 0), 3)
38     cv2.line(image, avatar_points['right_wrist'], avatar_points['right_hip'], (255, 0, 0), 3)
39     cv2.line(image, avatar_points['left_hip'], avatar_points['left_knee'], (255, 0, 0), 3)
40     cv2.line(image, avatar_points['right_hip'], avatar_points['right_knee'], (255, 0, 0), 3)
41     cv2.line(image, avatar_points['left_knee'], avatar_points['left_ankle'], (255, 0, 0), 3)
42     cv2.line(image, avatar_points['right_knee'], avatar_points['right_ankle'], (255, 0, 0), 3)
43     cv2.line(image, avatar_points['left_ankle'], avatar_points['left_hip'], (255, 0, 0), 3)
44     cv2.line(image, avatar_points['right_ankle'], avatar_points['right_hip'], (255, 0, 0), 3)
45     return image
46
47 while True:
48     success, image = cap.read()
49     if not success:
50         break
51     pose_landmarks = pose.solve(image)
52     if pose_landmarks is None:
53         continue
54     image = draw_avatar(image, pose_landmarks, image.shape[0], image.shape[1])
55     cv2.imshow('Avatar', image)
56     if cv2.waitKey(1) & 0xFF == ord('q'):
57         break
58     cv2.destroyAllWindows()
59
60 # Package list
61 packages: {
62   'dependencies': {
63     'angular/animations': '^15.2.10',
64     'angular/common': '^15.2.10',
65     'angular/core': '^15.2.10',
66     'angular/forms': '^15.2.10',
67     'angular/router': '^15.2.10',
68     'angular/service-worker': '^15.2.10',
69     'angular/ssr': '^15.2.10',
70     '@angular-devkit/build-angular': '^15.2.10',
71     '@angular-devkit/schematics': '^15.2.10',
72     '@angular/cli': '^15.2.10',
73     '@angular/compiler': '^15.2.10',
74     '@angular/compiler-cli': '^15.2.10',
75     '@angular/localize': '^15.2.10',
76     '@angular/platform-browser': '^15.2.10',
77     '@angular/platform-browser-dynamic': '^15.2.10',
78     '@angular/platform-server': '^15.2.10',
79     '@angular/pwa': '^15.2.10',
80     '@angular/ssr': '^15.2.10',
81     '@ngtools/webpack': '^15.2.10',
82     '@nguniversal/builders': '^15.2.10',
83     '@nguniversal/express-engine': '^15.2.10',
84     '@nguniversal/module-optimizer': '^15.2.10',
85     '@nguniversal/postcss-plugin': '^15.2.10',
86     '@nguniversal/router': '^15.2.10',
87     '@nguniversal/webpack-plugin': '^15.2.10',
88     '@nguniversal/webpack-plugin-express': '^15.2.10',
89     '@nguniversal/webpack-plugin-express-ssl': '^15.2.10',
90     '@nguniversal/webpack-plugin-express-ssl-https': '^15.2.10',
91     '@nguniversal/webpack-plugin-express-ssl-https-https': '^15.2.10',
92     '@nguniversal/webpack-plugin-express-ssl-https-https-https': '^15.2.10',
93     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https': '^15.2.10',
94     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https': '^15.2.10',
95     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https-https': '^15.2.10',
96     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https-https-https': '^15.2.10',
97     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https-https-https-https': '^15.2.10',
98     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https-https-https-https-https': '^15.2.10',
99     '@nguniversal/webpack-plugin-express-ssl-https-https-https-https-https-https-https-https-https-https': '^15.2.10',
100    }
101   }
102 }

```



ICT CAPACITY





MANUTENZIONE

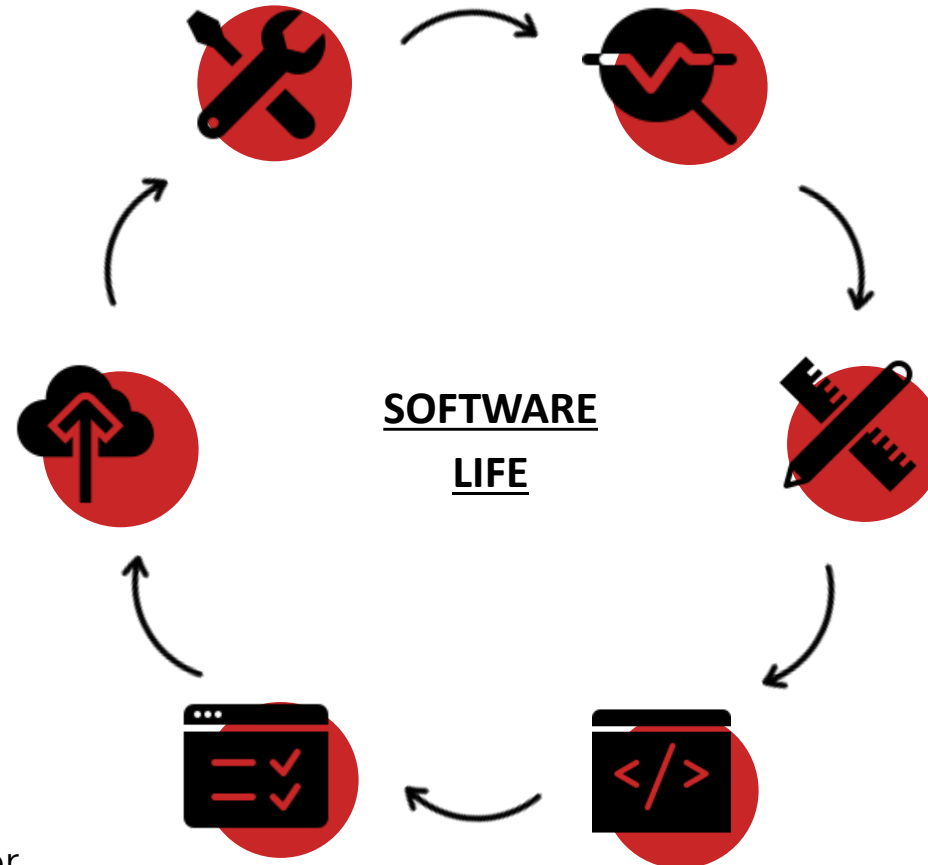
Maintenance Engineer
Technical Support
Data Analyst

DEPLOYMENT

DevOps Engineer
System Administrator
Release Manager

TESTING

QA Engineer
Test Automation Engineer



ANALISI

Business Analyst
Technical Consultant
Data Analyst

PROGETTAZIONE

Solution Architect
UX/UI Designer
Technical Lead

CODING

Software Developer
Frontend Developer
Backend Developer



HEALTHCARE

HEALTHCARE & BIOMEDICALE

- Program Management
- Know-how tecnico
- National Partnership
- Product development projects
- Develop joint project with Design, Engineering and ICT





THANK YOU

you desire. we design.