



# LAIRBOUND LEGENDS



## Tieto- ja viestintätekniikan Ohjelmisto 2 -kurssin projekti

Raise a New Dragon

Show My Dragon Stable

Through a rift in time and space, you were transported to the fantastical world of Eldaria, a world filled with dragons and elemental magic. Eldaria's scholars interpret this event as the fulfillment of an ancient prophecy, designating you as the new Dragonbound, tasked with raising the youth of the dragons, and guiding them through their journey to Dragonhood. The Dracarers, who are the caretakers and guardians of the dragon hatchlings in Eldaria, have decided that you will bond with one of their young Dragons, and together you will soar to the skies of destiny.

*Mira Peryshko, Jani Raitinen, Santeri Tamlander, Teemu Vataja* © G-Group Games



Hei! Olen säiden oraakkeli, ja tulin muistuttelemaan, mitä kaikkea näitkään demoesittelyssä!

- Jatkokehitysprojekti Lairbound Legendsistä, joka hyödyntää edellisen projektin toimintaympäristön lisäksi Frontendiä HTML:n, CSS:n ja JavaScriptin avulla.
- Kurssin tavoitteena oli syventyä olio-ohjelmointiin sekä saada käyttäjäympäristö keskustelemaan Backendin lisäksi API:en kanssa.
- Lisäksi peliin on sävelletty omaa musiikkia ja luotu grafiikkaa teokälyn avulla tai avoimien käyttöoikeuksien lähteistä.





Tässä on lähemmäs leikattu kuva  
pelin käyttäjänäkymästä! Keskiössä on  
Leaflet -kartta, ja oikealla pelin tilasta  
kertova papyrus-rolla.

Eldaria on ilmastokatastrofin jälkeinen todellisuus, jossa  
Helsinki-Vantaan Lair on pinnan alla veden-  
paisumuksen jäljiitä.

You are here: Helsinki Vantaa  
Lairport

Näitä painaessa sivu vaihtuu!



Spyro

♥ 100

🐛 10000



Tätä voi rullata alaspäin!

You have arrived at your destination  
Helsinki Vantaa Lairport.

Tässä lukee Backendin  
sanakirjaluokasta tuotavia  
tilannekatsauksia!



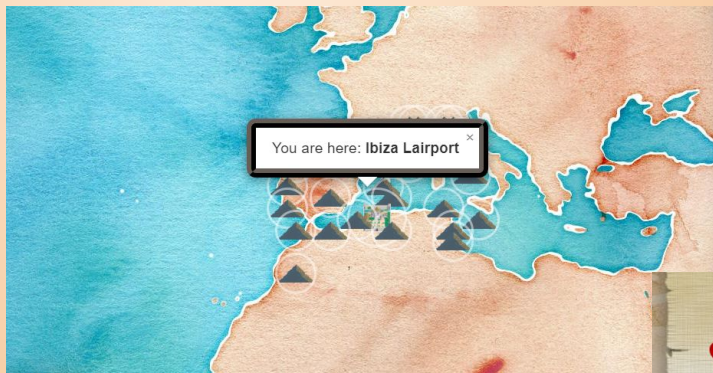
Tässä pari havainnekuvaa siitä, miten peli etenee ja mitä käyttöliittymästä näkee!



Papyruksen toiselta sivulta näkee tämänhetkisen sijainnin säätiedot!



Alempaa papyruksesta näkee arvoituksen siitä, missä säätilassa sinun ja lohikäärmeesi havittelema elemental shard sijaitsee!



Lohikäärmeen lentomatkan pituutta rajaa Stamina, ja lentäminen kuluttaa sitä!





# TAVERN



Ja kun et tiedä minne mennä, vieraillet minun tavernassani! Voit hörppiä potioneja kun statsit tippuvat, tai klikata minua ja käytän oraakkelisia voimiani etsiäkseni sinulle ilmansuunnan kohti oikeaa säätä. Odotas, näytän sinulle...



Potion of Health



Potion of Stamina

Ask help from the oracle of weather.  
He can sense where elemental forces gather.

Ask the oracle for guidance.



HNNNNNGHH  
H!!!

Ask help from the oracle of weather.  
He can sense where elemental forces gather

Let me gaze into the vortex, to find out where  
you should head next. Please be patient while I  
work.

```
def closestWeather(location, targetweather):
    # find closest of current weathergoals
    sql = "SELECT id, name, latitude_deg, longitude_deg FROM lairport"
    cursor = config.conn.cursor()
    cursor.execute(sql)
    result = cursor.fetchall()

    sql2 = "SELECT latitude_deg, longitude_deg FROM lairport WHERE id="
    cursor = config.conn.cursor()
    cursor.execute(sql2)
    location = cursor.fetchone()

    list = []
    target = None
    print(location)
    for i in result:
        entry = {}
        coo = {}
        if res[4] == "TEMP":
            # temperature rule
            dis = int(res[5]) <= temp <= int(res[6]):
                result = True
            ent
            ent
            ent
            lis
            sorted
        elif res[4] == "WIND":
            # wind rule
            if int(res[5]) <= wind <= int(res[6]):
                result = True
        for i in
        coo
        pri
        wea
        if
```

```
usage: <module>
def calculateDirection(location, targetweather):
    sql = "SELECT latitude_deg, longitude_deg FROM lairport WHERE id="
    cur = config.conn.cursor()
    cur.execute(sql)
    loc_coordinates = cur.fetchall()

    target_coordinates = closestWeather(location, targetweather) # the goal was to find
    # location coordinates
    lat1, lon1, lat2, lon2 = map(math.radians, [loc_coordinates[0], loc_coordinates[1], target_coordinates[0],
    target_coordinates[1]])
```

TADAA! Tarkemmat koodi-  
tiedot löytyvät GitHubista!

```
if weather:
    target = 1
    break
return target[1][2], target[1][3]
```

```
usage: <module>
def checkWeather(location, targetweather):
    apiKey = os.environ.get('API_KEY')

    request = "https://api.openweathermap.org/data/2.5/weather?lat=" + str(location[0]) + "&lon=" + str(location[1])
    vastaus = requests.get(request).json()
    main = vastaus["weather"][0]["main"]
    kelvin = vastaus["main"]["temp"]
    temp = int(kelvin - 273.15)
    wind = int(vastaus["wind"]["speed"])
    print(
        # longitude-tila
        d_lon = lon2 - lon1

    sql =
    cursor =
    y = math.sin(d_lon) * math.cos(lat2)
    cursor
    x = math.cos(lat1) * math.sin(lat2) - (math.sin(lat1)
    resu =
    print(
        # radianit
        bearing_rad = math.atan2(y, x)

        # radianit toastein asteiksi
        bearing_deg = math.degrees(bearing_rad)

        # muutetaan astetta 360-tilaan
        compass_point = (bearing_deg + 360) % 360

        # palautetaan astetta mukainen gitaarinen
        cardinal_directions = ["North", "Northeast", "East",
        index = round(compass_point / 45) % 8
        direction = ("direction": cardinal_directions[index])

    return direction
```

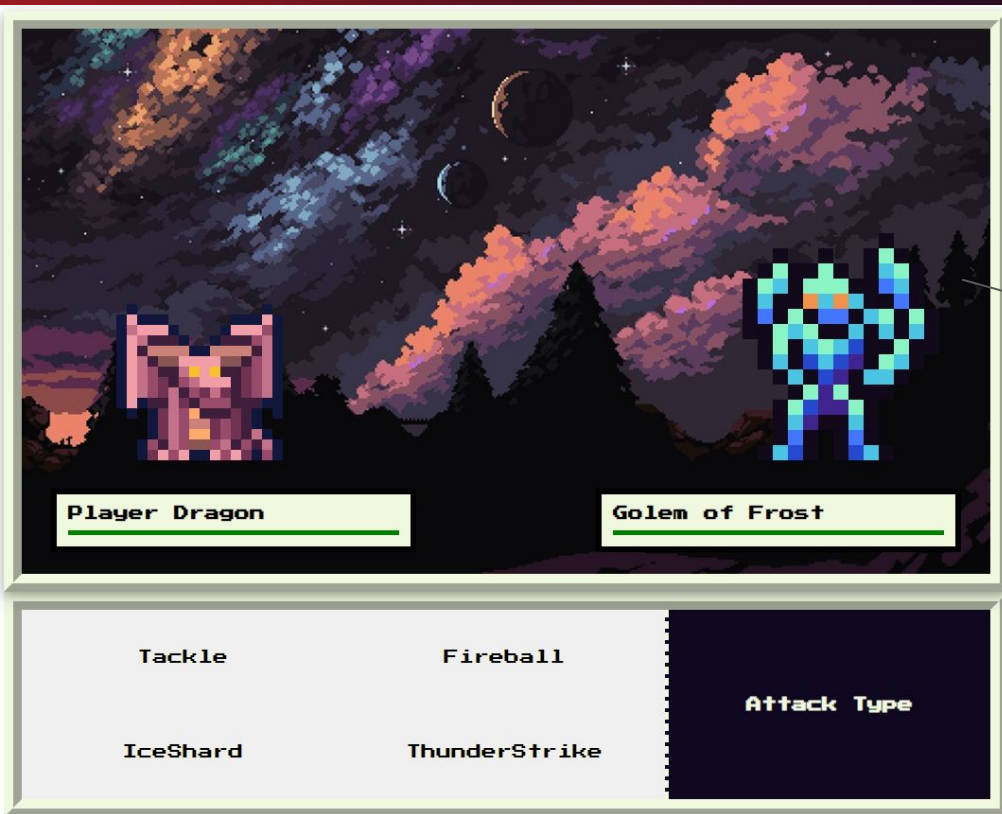


Ask help from the oracle of weather.  
He can sense where elemental forces gather.

Your fortune in **Northwest** I see.  
There is the lair where you need to be



# Lairbound Legends



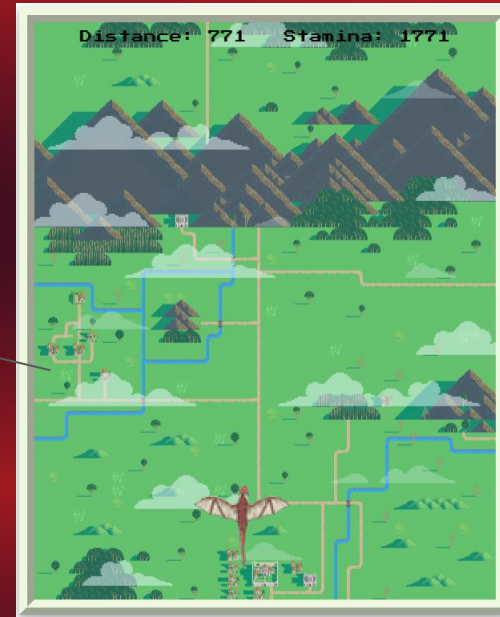
ther informa

Lopullinen versio tulee sisältämään kokonaan JavaScriptillä toteutetut lento- ja taistelumini-pelit. Koodien alut löytyvät jo myös GitHubista. Stay tuned!

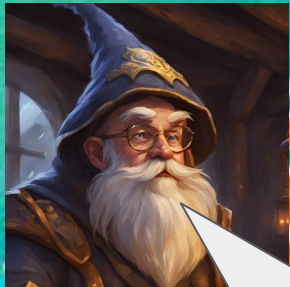


Päihitä vihollinen saadaksesi shardin!

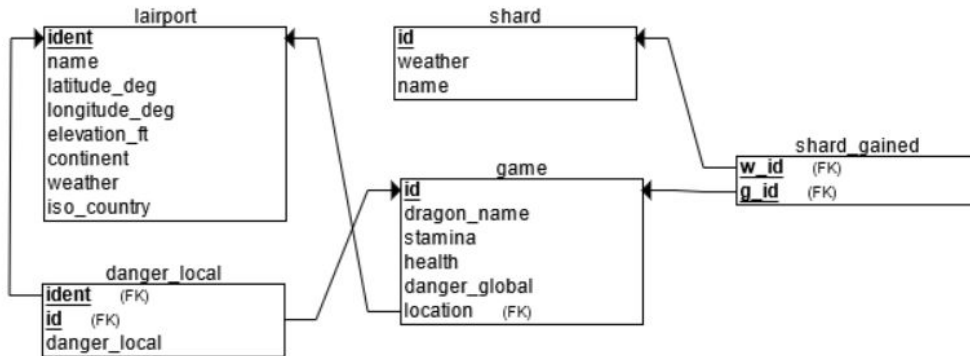
Väistele salmon snakejä selvitäksesi seuraavalle Lairille!



# Tietokanta



Danger\_local -taulua, ja game -taulun danger\_global arvoa ei käytetty tässä pelin 2. versiossa. Näidenkin osalta on käyttövisioita tulevaisuudessa!



Käytimme tietokantana projektin 1-osan tietokantaa, johon lisäsimme sääshardien keräämiseen tarvittavat kentät. Numeroimme shardit siten, että tietokannan numerot ovat samat kuin sanakirjassa, tietojenkäsittelyn helpottamiseksi.



## ***Rajapinnat ja linkit***

- Kartta: Leaflet (<https://leafletjs.com/>)
- kartan muotoilu: Stadia Maps (<https://stadiamaps.com/>)
- Säätilat: OpenWeatherMap (<https://openweathermap.org/>)
- Audio: Howler.js (<https://howlerjs.com/>)

### ***G-Group Games:***

***Santeri Tamlander***

***Mira Peryshko***

***Teemu Vataja***

***Jani Raitinen***



Tulevaisuudessa näen  
vielä paljon visioita pelin  
sisällöstä ja  
kehityssuunnasta!