

# Техникалық құжаттама



**Жоба тақырыбы: AI Fire Monitoring System**

**Команда атауы: NeuroSpace**

**Қатысушы: Абдіқадыр Нұрдәulet**

**Жетекші: Құлбаев Азамат Бектілеуұлы**

**Мектеп: №31 орта мектеп МҚМ**

**Қала/аудан: Алматы облысы, Іле ауданы, Жайнак ауылы**

## Кіріспе

AI Fire Monitoring System — жасанды интеллект пен деректерді талдау әдістерін қолдану арқылы орман және табиғи өрт ошақтарын ерте анықтауға, олардың таралу қаупін бағалауға және қауіпті деңгей жоғары болған жағдайда автоматты тұрде хабарлама жіберуге арналған интеллектуалды мониторинг жүйесі.

Қазіргі кезеңде орман өрттері әлемдік деңгейдегі ең өзекті табиғи апаптардың қатарына жатады. Жыл сайын миллиондаған гектар орман алқабы жаңып, бұл құбылыс экожүйелердің бұзылуына, биоалуантүрліліктің азауына, атмосфераға көмірқышқыл газының көп бөлінуіне және халық қауіпсіздігіне тікелей қатер төндіреді. Өрттердің басты мәселесі — олардың көп жағдайда кеш анықталуы; нәтижесінде өрт ауқымды аумаққа таралып, шығын көлемі еселенеді.

Осы проблеманы шешуге бағытталған AI Fire Monitoring System жобасы төмендегі негізгі функцияларды орындаиды:

- әлемдік деңгейде өрт ошақтарын анықтау және бақылау;
- әрбір өрт нүктесінің қауіптілік деңгейін есептеу;
- интерактивті картада визуализациялау;
- жоғары қауіпті жағдайлар бойынша Telegram арқылы жедел хабарлама жіберу.

Жүйе үлкен көлемдегі деректерді өндөуге бейімделген және 24/7 режимінде үздіксіз жұмыс істеуге арналған.

---

## Жобаның мақсаты

Жобаның негізгі мақсаты — жасанды интеллект және деректер аналитикасы негізінде табиғи өрттерді ерте анықтайтын, оларды қеңістікте бақылайтын және қауіпті жағдайларда жедел ескерту жасайтын автоматтандырылған мониторинг жүйесін әзірлеу.

---

## Жобаның міндеттері

**Жоба келесі міндеттерді қамтиды:**

- өрт ошақтарының нүктелерін анықтау және деректерді жинау;
- өрт қауіптілігін сандық көрсеткіш арқылы бағалау;
- өрт нүктелерін интерактивті картага шыгару;
- жоғары қауіпті оқиғаларға автоматты хабарландыру жүйесін іске қосу;
- үлкен деректерді өндедеу және статистикалық талдау жүргізу.

Бұл міндеттерді орындау өрттің алдын алу шараларын күшейтіп, таралу аймағын азайтуға мүмкіндік береді.

---

### Жобаның өзектілігі

Орман өрттері жаһандық экологиялық әрі экономикалық проблема ретінде қарастырылады. Кейбір жағдайларда өрт оқиғалары бірнеше аптаға созылып, аса үлкен аумақты қамтиды. Өрттің жиілеуіне әсер ететін негізгі факторлар:

- климаттың өзгеруі және температураның көтерілуі;
- құрғақшылық пен ылғал тапшылығы;
- найзағай разрядтары;
- адам факторлары (техникалық қауіпсіздік бұзылуы, абайсыздық, қасақана әрекеттер).

Қазіргі мониторинг жүйелерінде жиі кездесетін кемшіліктер:

- үлкен аумақты толық қамту қындығы;
- деректердің кеш өнделуі;
- өрт туралы ақпараттың уақытылы жетпеуі.

Осы тұста AI-негізіндегі жүйелердің артықшылығы анық байқалады: олар үлкен деректерді жылдам өндейді, қауіпті жағдайларды автоматты таниды және болжau-талдау жүргізуге мүмкіндік береді. Сондықтан AI Fire Monitoring System секілді шешімдердің практикалық маңызы жоғары.

---

### Жобаның қысқаша сипаттамасы

AI Fire Monitoring System — веб-платформа түріндегі мониторинг жүйесі. Оның негізгі мүмкіндіктері:

- 10 000+ өрт нүктесін бақылау мүмкіндігі;
- интерактивті карта интерфейсі;

- AI негізінде қауіптілік деңгейін есептеу;
- Telegram арқылы автоматты ескерту (alert) жүйесі;
- статистика және аналитикалық есептер.

Жоба Python бағдарламалау тілінде жүзеге асырылған.

---

## Жүйе архитектурасы

Жүйе бірнеше функционалдық модульден тұрады:

### 1) Data Generation Module (Деректерді жинау/генерациялау модулі)

Бұл модуль өрт нүктелері бойынша деректерді генерациялайды немесе сыртқы деректер көздерінен алады. Негізгі параметрлер:

- Latitude — географиялық ендік;
- Longitude — географиялық бойлық;
- Brightness — жылу интенсивтілігі;
- Confidence — анықтау сенімділігі.

Аталған деректер өрт ошағының орнын және сипаттамаларын анықтауға негіз болады.

### 2) Data Processing Module (Деректерді өндіу модулі)

Модульдің міндеті — деректердің сапасын арттыру және талдауға дайындау:

- деректерді тазалау;
- қате/шұлы мәндерді сұзу;
- құрылымдау және форматтау;
- статистикалық талдау жүргізу.

Бұл кезең жүйенің тұрақты әрі сенімді жұмыс істеуін қамтамасыз етеді.

### 3) AI Risk Analysis Module (Қауіптілікті бағалау модулі)

Өрт қаупі Risk көрсеткіші арқылы есептеледі. Қолданылатын формула:

$$\text{Risk} = (\text{Brightness} / 400) \times 0.6 + (\text{Confidence} / 100) \times 0.4$$

Мұнда екі негізгі фактор біріктіріледі:

1. жылу интенсивтілігі (Brightness);
2. анықтау сенімділігі (Confidence).

Нәтижесінде әрбір өрт нүктесіне сандық қауіптілік деңгейі беріледі.

#### 4) Visualization Module (Визуализация модулі)

Өрт нүктелері интерактивті картада түстер арқылы көрсетіледі:

- ● төмен қауіп;
- ○ орташа қауіп;
- ● жоғары қауіп.

Бұл тәсіл пайдаланушыға жағдайды жылдам бағалауға мүмкіндік береді.

#### 5) Notification Module (Хабарландыру модулі)

Жоғары қауіпті өрттер бойынша жүйе Telegram арқылы автоматты хабарлама жібереді. Хабарлама құрамында:

- өрт тіркелген ел/аймақ;
- координаталар;
- қауіптілік деңгейі (Risk);
- Brightness және Confidence көрсеткіштері.

Бұл функция жедел әрекет ету уақытын қысқартады.

---

### Қолданылған технологиялар

Жүйені құрастыруда келесі технологиялар қолданылды:

- Python — негізгі бағдарламалу тілі (AI, деректер талдау, веб-қосымша әзірлеу үшін);
  - Streamlit — веб-интерфейс пен қолданба жасау;
  - Pandas — деректерді өндөу, құрылымдау және статистика;
  - NumPy — математикалық есептеулер;
  - Folium — интерактивті карта және геодеректер визуализациясы;
  - Telegram Bot API — автоматты хабарлама жіберу.
-

## Жүйенің жұмыс алгоритмі

Жүйенің жұмыс істеу тәртібі:

1. өрт деректерін жинау;
2. деректерді алдын ала өңдеу;
3. AI арқылы қауіптілік деңгейін есептеу;
4. интерактивті картада көрсету;
5. жоғары қауіпті оқиғаларды сұзу;
6. Telegram арқылы хабарлама жіберу.

Бұл процесс толықтай автоматтандырылған.

---

### Жүйені іске қосу қадамдары (басы)

**1-қадам:** PyCharm немесе басқа Python ортасын (IDE) орнату және іске қосу.

```
2-қадам: import streamlit as st
import pandas as pd
import numpy as np
import folium
from folium.plugins import MarkerCluster
from streamlit_folium import st_folium
import requests
import time
import math

# =====
# TELEGRAM CONFIG
TELEGRAM_TOKEN = "8440286651:AAFhJmqBLuWQpz7oRV3vPwxJYxvViCzgmYE"
TELEGRAM_CHAT_ID = "5977636479"

def send_telegram(text):
    url = f"https://api.telegram.org/bot{TELEGRAM_TOKEN}/sendMessage"
    try:
        requests.post(url, data={"chat_id": TELEGRAM_CHAT_ID, "text": text},
                      timeout=5)
        time.sleep(0.1)
    except:
        pass

# =====
# FIRE DATA GENERATION (10k points)

@st.cache_data(ttl=600)
def generate_fire_data(n=10000):

    np.random.seed(42)

    df = pd.DataFrame({
        "latitude": np.random.uniform(-50, 60, n),
        "longitude": np.random.uniform(-180, 180, n),
        "brightness": np.random.uniform(300, 500, n),
```

```

        "confidence": np.random.uniform(50, 100, n),
        "wind_speed": np.random.uniform(0, 15, n),
        "wind_direction": np.random.uniform(0, 360, n)
    })

df["risk"] = (df["brightness"]/400)*0.6 + (df["confidence"]/100)*0.4

return df

# =====
# AI FIRE SPREAD MODEL

def predict_spread(lat, lon, wind_speed, wind_dir):

    distance = wind_speed * 0.01

    rad = math.radians(wind_dir)

    new_lat = lat + distance * math.cos(rad)
    new_lon = lon + distance * math.sin(rad)

    return new_lat, new_lon

# =====

st.set_page_config(page_title="🔥 AI Fire Monitoring", layout="wide")

st.title("🔥 Global AI Fire Monitoring System")

df = generate_fire_data()

# =====
# MAP

st.subheader("📍 Fire Map")

m = folium.Map(location=[20,0], zoom_start=2)

cluster = MarkerCluster().add_to(m)

high_risk = []

for _, row in df.iterrows():

    lat = row["latitude"]
    lon = row["longitude"]

    risk = row["risk"]

    if risk > 0.7:
        color = "red"
    elif risk > 0.4:
        color = "orange"
    else:
        color = "green"

    folium.CircleMarker(
        location=[lat, lon],
        radius=3,
        color=color,
        fill=True,
        fill_opacity=0.7,
        popup=f"Risk:{risk:.2f}"
    )

```

```
    ).add_to(cluster)

    # AI spread prediction
    new_lat, new_lon = predict_spread(
        lat,
        lon,
        row["wind_speed"],
        row["wind_direction"]
    )

    folium.CircleMarker(
        location=[new_lat, new_lon],
        radius=2,
        color="blue",
        fill=True
    ).add_to(cluster)

    if risk > 0.8:
        high_risk.append(row)

st_folium(m, width=1300, height=700)

# =====
# AI STATISTICS

st.subheader("▣ AI Fire Prediction")

st.write("Total fires:", len(df))
st.write("High risk fires:", len(high_risk))

# =====
# AUTO TELEGRAM ALERT

if len(high_risk) > 0:

    send_telegram(f"⚠ HIGH RISK FIRES DETECTED: {len(high_risk)}")

    for row in high_risk[:20]:

        msg = f"""
🔥 HIGH RISK FIRE

Lat: {row['latitude']:.3f}
Lon: {row['longitude']:.3f}

Risk: {row['risk']:.2f}
Brightness: {row['brightness']:.1f}
Confidence: {row['confidence']:.1f}

Wind Speed: {row['wind_speed']:.1f}
Wind Direction: {row['wind_direction']:.0f}
"""

        send_telegram(msg)

st.success("⚡ System monitoring fires 24/7") - осы кодты IDLE-ге көшіріп қою.
```

## Жүйені іске қосу қадамдары

### 3-қадам: Қажетті кітапханаларды орнату

Жүйенің дұрыс жұмыс істеуі үшін Python ортасына қажетті кітапханалар орнатылады. Ол үшін командалық жолға (Command Prompt / Terminal) келесі команданы енгізу қажет:

```
pip install streamlit pandas numpy folium scikit-learn requests
```

Бұл кітапханалар веб-интерфейс құруға, деректерді өндөуге, математикалық есептеулер жүргізуге, картада визуализация жасауға және сыртқы сұраныстарды орындауға мүмкіндік береді.

### 4-қадам: Streamlit қосымшасын іске қосу

Орнату аяқталғаннан кейін веб-қосымшаны іске қосу үшін командалық жолға төмендегі команданы енгіземіз:

```
streamlit run (файл_немесе_папка_атауы)
```

Команда орындалғаннан кейін Streamlit сервері автоматты түрде іске қосылып, жүйе браузер арқылы сайт ретінде ашылады.

### 5-қадам: Telegram арқылы автоматты хабарлама алу

Жоғары қауіпті өрт нүктелері анықталған жағдайда жүйе Telegram Bot арқылы автоматты түрде хабарлама (SMS түріндегі мәтіндік ескерту) жібереді. Бұл үшін арнайы бот жасалады және жүйеге Telegram Bot API токені енгізіледі.

Мысалы, бот атауы: Fire Monitoring Bot  
Хабарлама құрамында қауіпті өрт туралы негізгі мәліметтер (орналасуы, координаталары, қауіп деңгейі және негізгі көрсеткіштер) беріледі.

---

## Жүйенің артықшылықтары

AI Fire Monitoring System бірнеше маңызды артықшылықтарға ие:

- Үлкен көлемдегі деректерді жылдам өндөу мүмкіндігі;
- қауіпті жағдайлар бойынша автоматты хабарландыру жүйесі;

- интерактивті карта арқылы көрнекі бақылау;
- жасанды интеллект негізіндегі тәуекелді талдау;
- 24/7 үздіксіз мониторинг режимі.

Аталған мүмкіндіктер жүйеге өрт ошақтарын ерте анықтап, шұғыл әрекет ету үақытын қысқартуға жағдай жасайды.

---

### **Болашақтағы жетілдірүлөр**

Жүйені әрі қарай бірнеше бағытта дамытуға болады. Болашақта енгізілетін ықтимал жетілдірүлөр:

- нақты спутниктік деректер көздерін жүйеге интеграциялау;
- машиналық оқыту алгоритмдерін жетілдіру және модель дәлдігін арттыру;
- өрттің таралу бағытын және жылдамдығын болжау модулін енгізу;
- мобиЛЬДІ қосымша нұсқасын өзірлеу;
- жүйені cloud серверге орналастырып, тұрақты онлайн қызметке шығару.

Бұл жаңартулар жобаны толыққанды кәсіби деңгейдегі мониторинг платформасына айналдыруға мүмкіндік береді.

---

### **Қорытынды**

AI Fire Monitoring System — орман және табиғи өрттерді анықтау мен бақылауға арналған интеллектуалды мониторинг жүйесі. Жүйе:

- өрт нүктелерін анықтайды;
- қауіптілік деңгейін есептейді;
- интерактивті картада визуализациялады;
- жоғары қауіп кезінде автоматты түрде хабарлама жібереді.

Осы технология табиғи апаттардың алдын алуда және өрт салдарын азайтуда тиімді қолданылатын маңызды цифрлық құрал болып табылады.