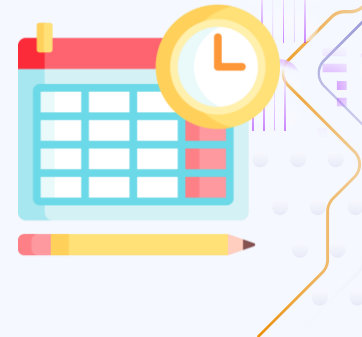




UNIVERSITATEA „DUNĂREA DE JOS” DIN GALAȚI  
FACULTATEA DE AUTOMATICĂ, CALCULATOARE, INGINERIE  
ELECTRICĂ ȘI ELECTRONICĂ



# APLICAȚIE PENTRU PLANIFICARE INTELIGENTĂ UTILIZÂND TEHNICI DE AI

Coordonator științific,  
**Conf.dr.ing. Mihaela ȚIPLEA**

Absolvent,  
**Liliana- Petruța ZANFIR**

Galați  
2025



# Structura lucrării

1.

Scopul și obiectivele proiectului

2.

Descrierea aplicației și a problemei abordate

3.

Tehnologii utilizate

4.

Arhitectura, funcționarea și metodologia aplicație

5.

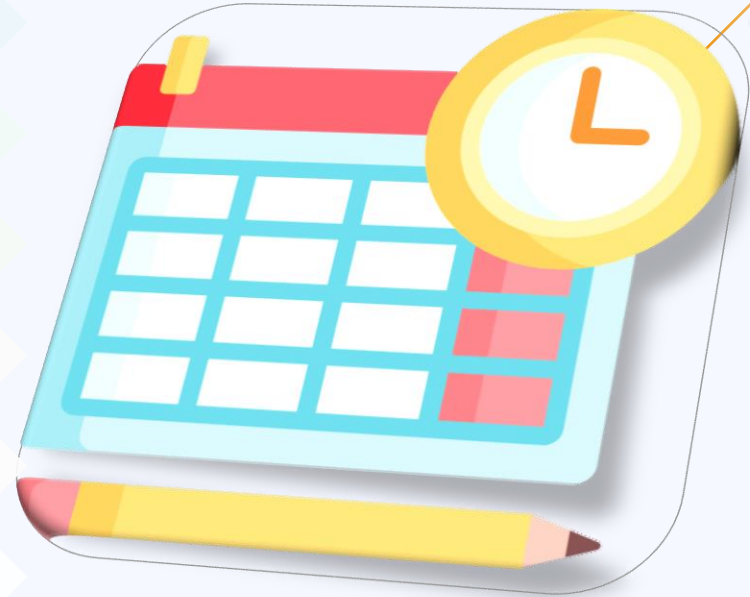
Rezultate obținute

6.

Comparații AI vs algoritm clasic

7.

Concluzii și perspective de dezvoltare



# Scopul și obiectivele lucrării



## Pentru ce?

Automatizarea procesului  
de generare a orarului  
universitar



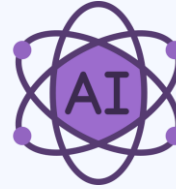
## De ce?

Reducerea timpului și a  
erorilor umane în  
planificare



## Cum?

Integrarea unei soluții AI  
comparativ cu un algoritm clasic



## Sub ce formă?

Oferirea unei aplicații web  
moderne și scalabile



# Descrierea aplicației și a problemei abordate

Necesitatea automatizării și planificării academice



Soluție propusă: aplicație web modernă pentru generarea automată a orarului



Problemele orarelor realizate manual



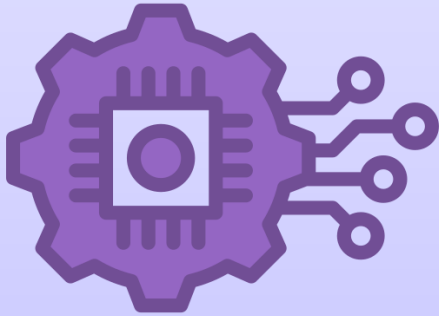
Generare orar cu AI și algoritmi clasici



Gestionare completă: profesori, săli, grupe, reguli



# Tehnologii utilizate și metode de generare



## Frontend:

React + Bootstrap (UI modern, responsive)



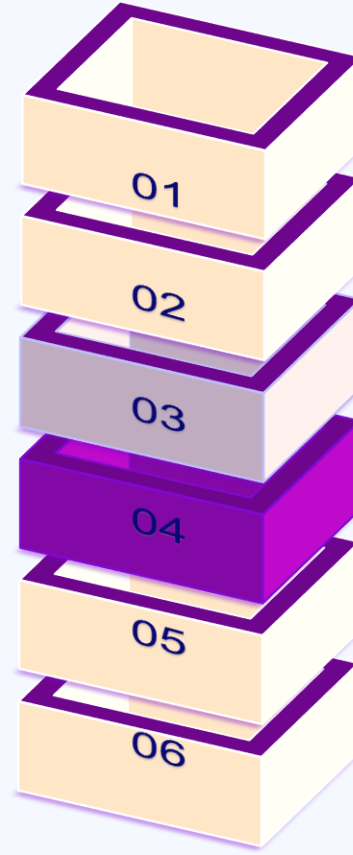
## Backend:

Flask (Python) — API REST pentru logica aplicației



## Bază de date:

MySQL — stocare profesori, săli, grupe, reguli, orare



**Generare orar AI:**  
OpenAI GPT-4



**Generare orar Clasic:**  
algoritm Python cu reguli proprii



## Export:

PDF (ReportLab, PyPDF2) și Excel (xlsxwriter)

React (UI) → Flask (API) →  
MySQL (DB)

Generare orar fie cu AI  
(GPT-4), fie cu algoritm  
clasic Python

Introducere date →  
generare orar →  
validare automată →  
export PDF/Excel

Structură JSON și reguli  
academice aplicate  
automat

**Arhitectură  
pe 3 niveluri**

**Metodologie**

**Flux principal**

**Validare**



**Arhitectura,  
funcționarea și  
metodologia aplicație**



# Rezultate obținute

Reducere semnificativă a timpului  
față de planificarea manuală

Testare funcțională și comparativă  
pentru ambele metode

1

Generare automată de orare conforme cu  
toate regulile academice

2

3

Validare automată a structurii și  
restricțiilor

4

Export rapid în PDF și Excel, gata de utilizat

5

# Comparație AI vs algoritm clasic

Aspect	AI (GPT-4 / Mistral)	Algoritm clasic Python
Flexibilitate	● ● ● ● ● adaptare instantă	● ● modificări manuale
Timp execuție	● ● ● ● rapid	● ● ● ● similar, static
Adaptabilitate	● ● ● ● ● scalabil, învață	● rigid, greu de extins
Mentenanță	● ● ● ● simplu	● ● necesită rescriere



# Rezultate

Automatizarea  
generării orarului  
universitar, eliminând  
procese manuale.



Economie semnificativă  
de timp și eliminarea  
erorilor umane.



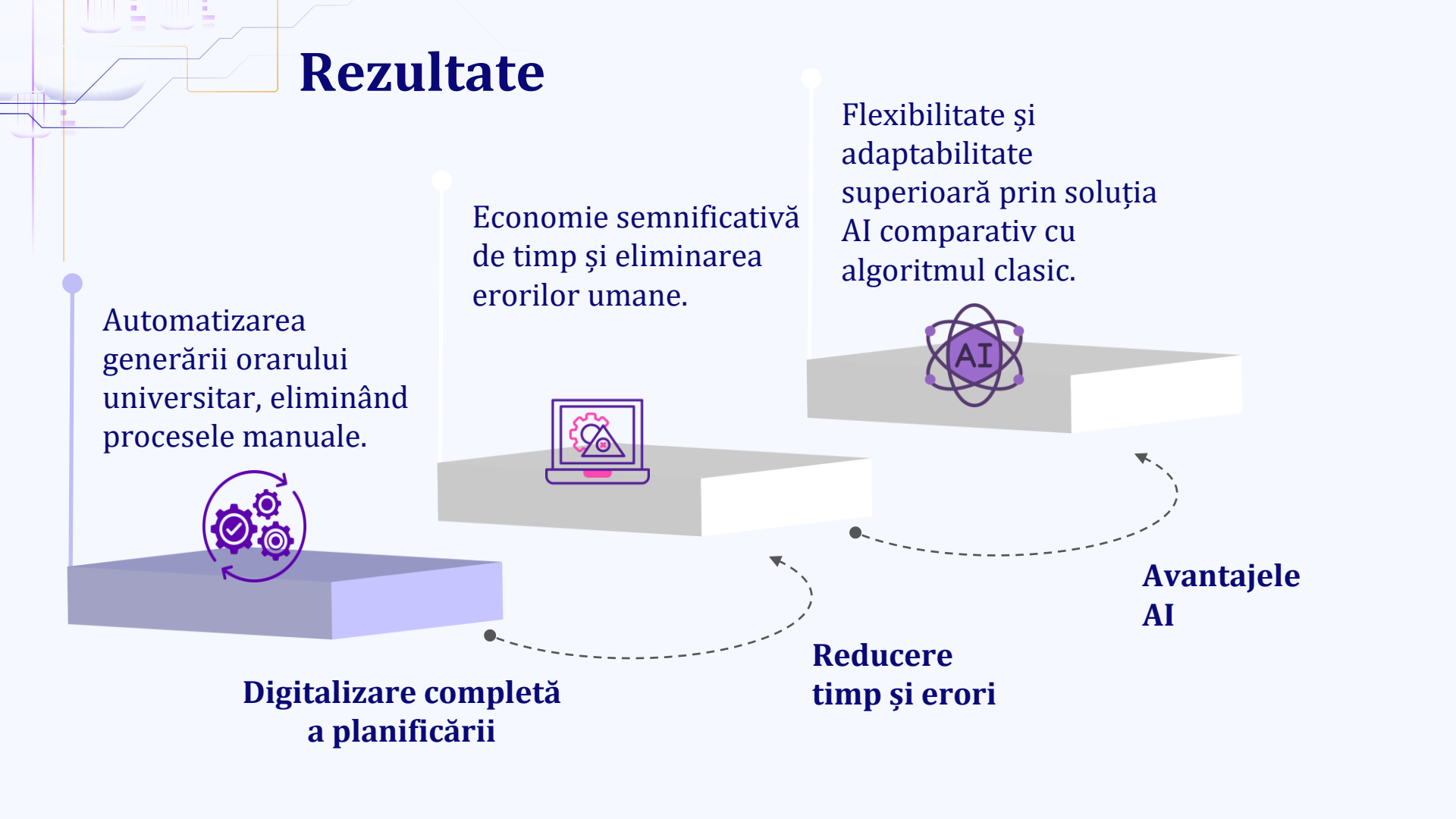
Flexibilitate și  
adaptabilitate  
superioară prin soluția  
AI comparativ cu  
algoritmul clasic.



**Digitalizare completă  
a planificării**

**Reducere  
timp și erori**

**Avantajele  
AI**

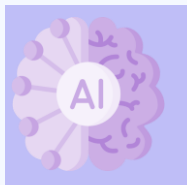


# Perspective de dezvoltare



## Extinderea funcționalităților aplicației

- profiluri utilizatori (admin, profesor, student)
- notificări automate



## Optimizare tehnică și AI

- testare modele AI locale
- generare offline



## Adaptare și mobilitate

- orare pentru examene / extra
- aplicație Android / iOS



„Success is the sum of small efforts, repeated day in and day out.”

— Robert Collier



**Vă mulțumesc pentru atenție!**