SOLID Principles & Clean Code

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com

AGENDA

1. Why and what is S.O.L.I.D
2. S.O.L.I.D. deep dive
3. Bonus clean code principles
4. Clean code guidelines
5. References

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com

UNIUNEA EUROPEANĂ

Instrumente Structurale
2014-2020

Why and what is S.O.L.I.D

- Are a set of rules that enable developers to write software that is easily extendable, maintainable easy to read and easy to adapt to changing requirements
- If we don’t conform to these standards the code can become inflexible and more brittle, small changes in the software can result in bugs

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com

UNIUNEA EUROPEANĂ

Instrumente Structurale 2014-2020

SINGLE RESPONSIBILITY PRINCIPLE

How to identify

- A class or method has more than one responsibility

Good

- Less complexity(one thing per class, more readable, easy to test)
- Methods become highly related (work together to do one thing well - coherent)
- Classes less dependent on each other(decupled)

Downsides

- What is "one reason to change"? Where to draw the line?
- In legacy projects is difficult to implement, takes time
- Explosion of tiny methods, take in consideration better organizing and grouping of files

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com

OPEN CLOSE PRINCIPLE

How to identify

- A class or function is always open for modification, in other words always allows adding more logic to it

Good

- Lower chances to introduce bugs
- Only test and deploy new features

Downsides

- Sometimes just can't be applied and you need to modify existing functionality
- When that happens try to make as few changes as possible

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com

UNIUNEA EUROPEANĂ
Instrumente Structurale 2014-2020

LISKOV SUBSTITUTION PRINCIPLE

How to identify

- CREATE A BEHAVIOR THAT IS NOT SPECIFIED IN THE BASE CLASS
- USE UNEXPECTED BEHAVIORS THAT DO NOT POSE AN ISSUE AT RUNTIME, BUT DON'T WORK FUNCTIONALLY

Good

- Keep functionality intact and still be able to use a subclass as a substitute for a base class
- Code reusability
- Prevents problems emerging at runtime but rather at build time

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com

INTERFACE SEGRAGATION PRINCIPLE

How to identify

- Classes MUST IMPLEMENT PARTS OF INTERFACE THAT THE NOT NEED
- WE PASS NULL (OR SIMILAR) AS A PARAMETER
- METHOD THROWS NOT IMPLEMENTED EXCEPTION

Good

- A client is not forced to implement parts of interfaces that are not used
- Easier to implement
- Methods are highly related to each other
- Better readability and maintainability
- Easier to test, not need to mock so much

OBSERVATIONS

- By violating this we also violate single responsibility principle

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com

DEPENDENCY INVERSION PRINCIPLE

Good

- Easy to swap implementations
- Easy to test (mock dependencies you don't care of)
- Avoid using new in code
- Don't be tight coupled to other classes
- Parallel development is possible
- Can use DI
- Can use a container to register all dependencies

Observations

- Use interfaces
- Dependency inversion is the principle
- Dependency injection is used to make the principle work

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com

BONUS CLEAN CODE PRINCIPLES

- DRY
- Don’t Repeat Yourself

KISS

- Keep It Simple Stupid/ Keep It Stupid Simple

YAGNI

- You Aren’t Gonna Need It

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com

UNIUNEA EUROPEANĂ

Instrumente Structurale
2014-2020

CLEAN CODE GUIDELINES

- Follow S.O.L.I.D Principles
- Code should readable (spacing, formatting)
- Code should be easy to understand (naming conventions)
- Code should be well structured (solutions, namespaces, folders)
- Code should be highly testable, extendable and easy to maintain
- Be consistent
- Avoid comments
- Try to write code that unfolds like a story when is read

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com

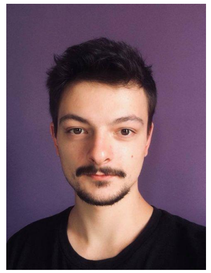
REFERENCES

- Clean Code A Handbook of Agile Software Craftsmanship by **Robert C. Martin** (added with the presentation)

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com

Thank you!



Mirel Aioanei

Mirel.Aioanei@endava.com

Cristian Tugui

Stefan.Tugui@endava.com

Proiect cofinantat din Fondul Social European prin Programul Operaţional Capital Uman 2014-2020

Endava Romania SRL
Str. Al. Vaida Voevod Nr. 51
400436 Cluj-Napoca, România
T +40 372 363 282
F +40 264 429 027
endava.com