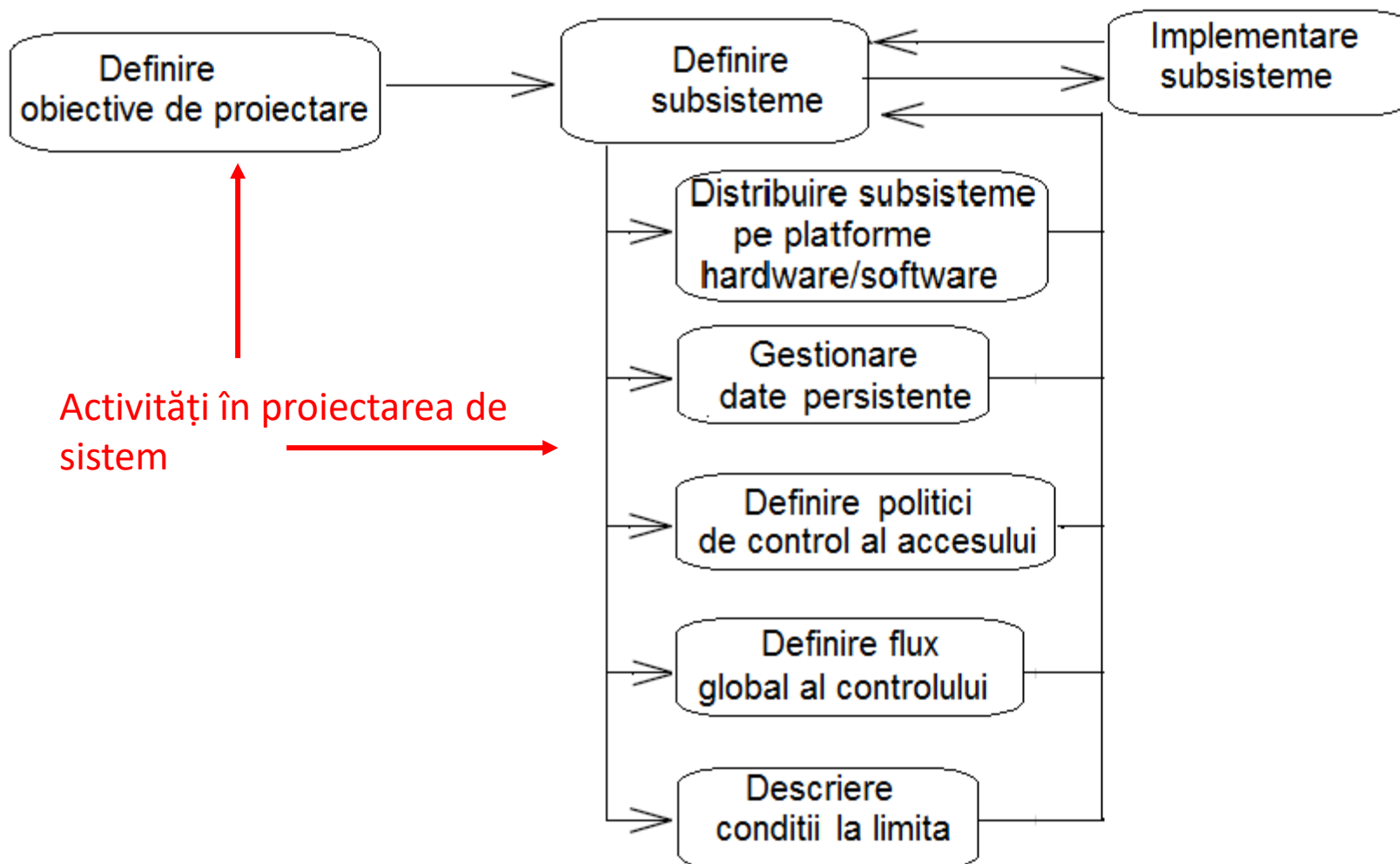


Proiectarea arhitecturala -3

Prof. univ. dr. ing. Florica Moldoveanu

Curs *Ingineria programelor* – UPB, Automatică și Calculatoare
2020-2021

Activități în proiectarea de sistem



Identificarea condițiilor limita (boundary conditions)

- Proiectare de sistem considera, în principal, **comportamentul sistemului în starea sa stabila.**
- Este necesar sa se examineze și **conditiile limită în functionarea sistemului:**
 - **pornirea, initializarea, oprirea**
 - **căderile majore** care pot duce la pierderea datelor, întreruperile în comunicatia prin retea, etc.
 - cauzate de:
 - un defect software
 - o greseala de operare
 - o întrerupere a sistemului electric, etc.
 - **atacuri la securitatea sistemului**
 - **conditii de utilizare pentru administrarea sistemului**
- Functionarea sistemului în aceste situatii este descrisa prin **cazuri de utilizare limită sau administrative (boundary use cases/administrative use cases)** – deoarece actorul care interactioneaza cu sistemul in aceste cazuri de utilizare este de regula administratorul de sistem

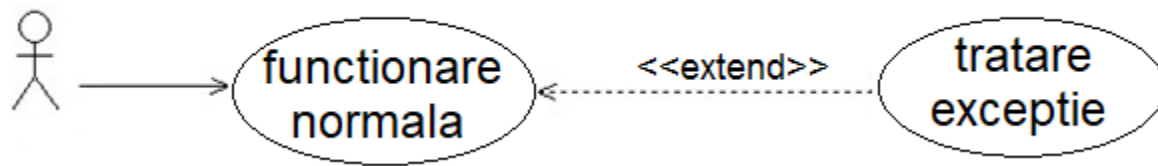
Cazuri de utilizare în condiții limită

Pornirea, oprirea si configurarea

- Pentru fiecare componenta executabila a sistemului se adauga 3 cazuri de utilizare: pornirea, oprirea si configurarea (initializarea) componentei.

Tratarea exceptiilor

- Anumite exceptii pot fi tolerate de sistem și incluse în proiectarea unor componente.
- Pentru fiecare tip de **cădere prevazuta** se scrie un caz de utilizare care extinde unul dintre cazurile de utilizare descrise in faza de extragere a cerintelor.
- **Modul de tratare a acestor exceptii este decis in faza de proiectare de detaliu.**



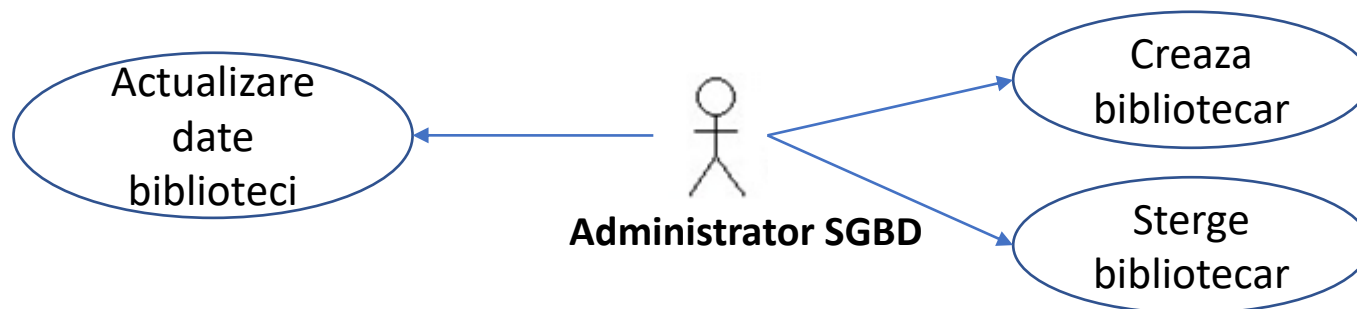
Cazuri de utilizare administrative

- Pentru operatii de administrare a sistemului, orice aplicatie ar trebui sa ofere **o interfata separata** de aceea care permite utilizatorilor accesul la functiile sistemului.
- De ex., **pentru o aplicatie web se recomanda** ([OWASP Guide 2.0](#)) **sa existe o aplicatie separata, dedicata sarcinilor administrative**, care sa poata fi accesata numai printr-un sistem de management de retea care asigura securitatea comunicarii. De ex, navigarea printr-o retea VPN (Virtual Private Network) este sigura deoarece traficul este criptat.

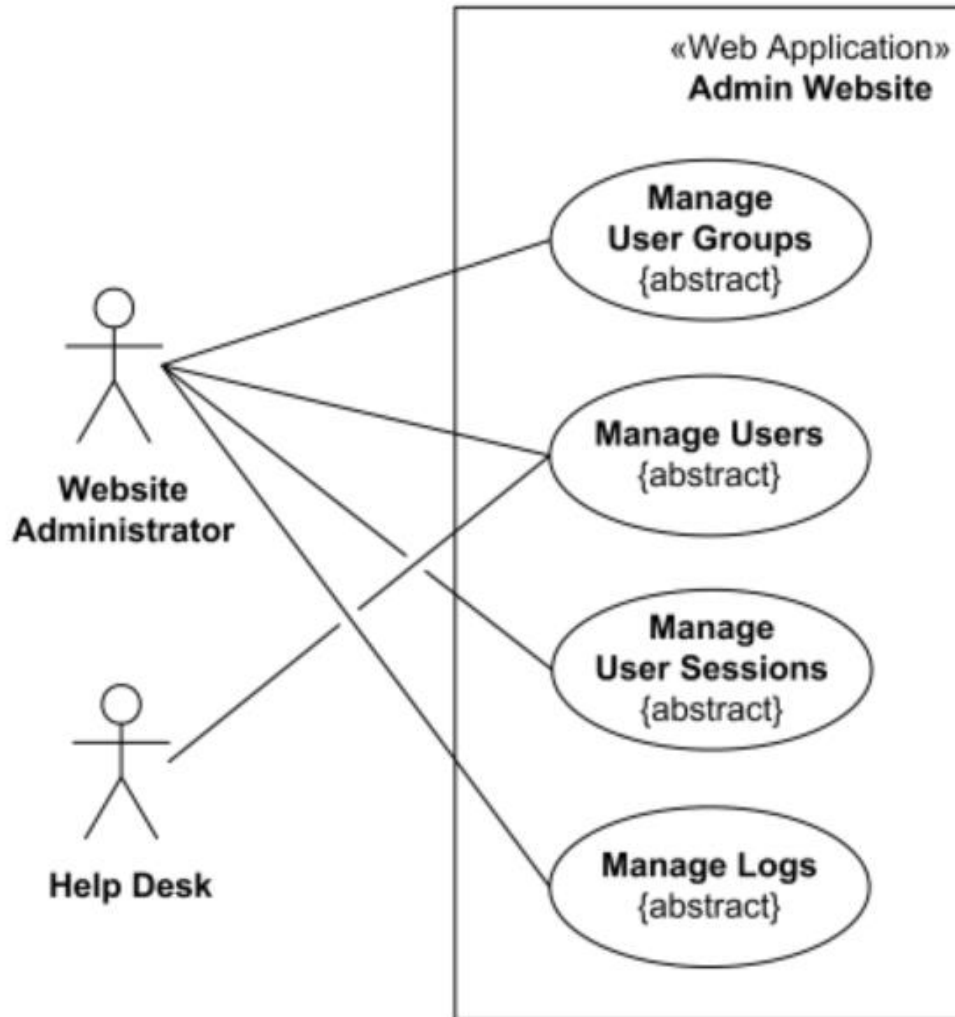
Exemplu – SGCB

Bibliotecarul gestioneaza cartile din biblioteci: adauga, sterge, modifica nr. de exemplare.

Pentru accesul la sistem are nevoie de un cont creat de administratorul sistemului.



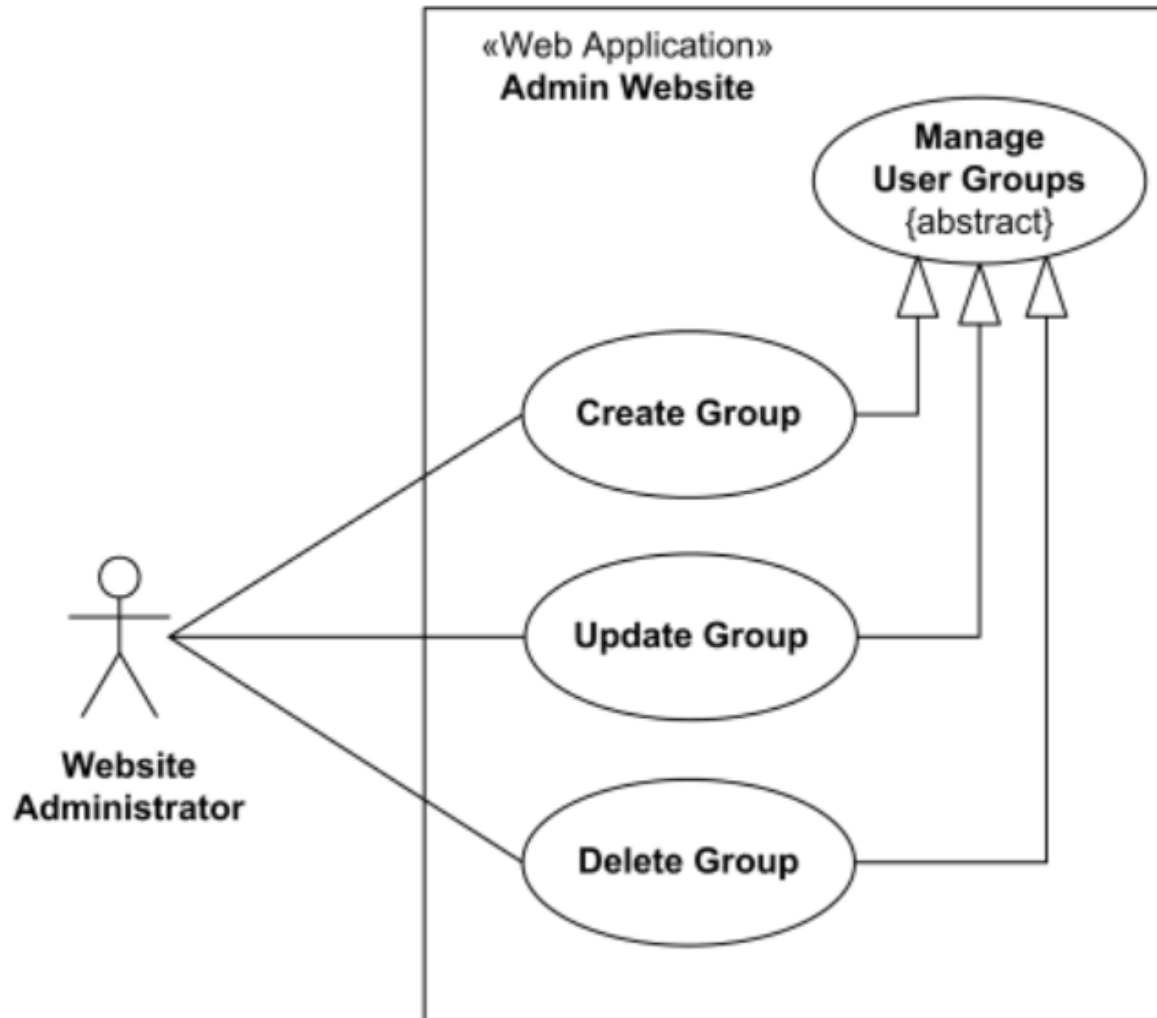
Exemplu: diagrama de cazuri de utilizare de nivel inalt pentru administrarea unui site web



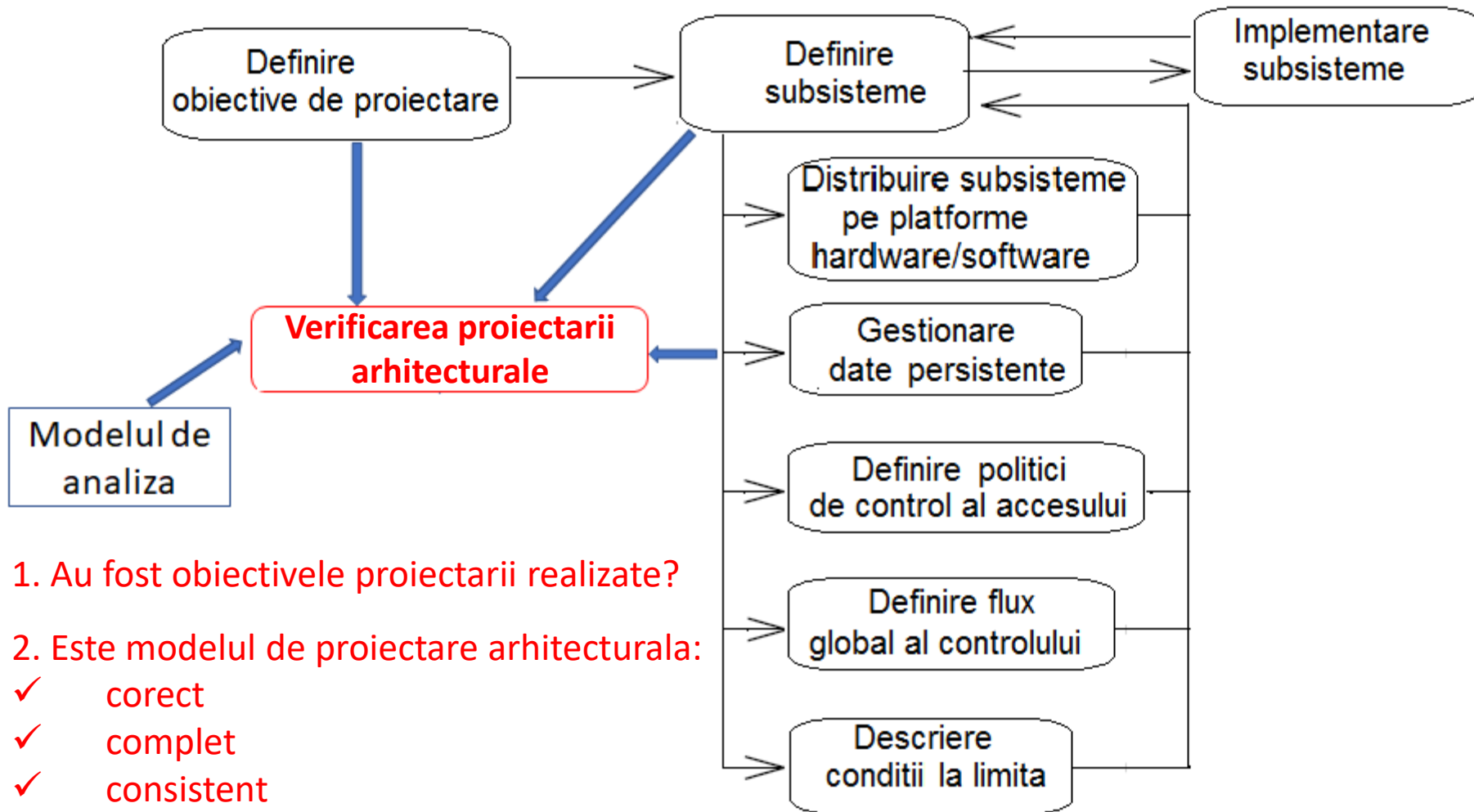
Caz de utilizare abstract:

- Neinstantiabil
- Poate fi un CU care:
 - Este inclus in alt CU
 - Extinde un alt CU
 - Generalizeaza alte CU

Exemplu - continuare



Verificarea proiectarii arhitecturale (1)



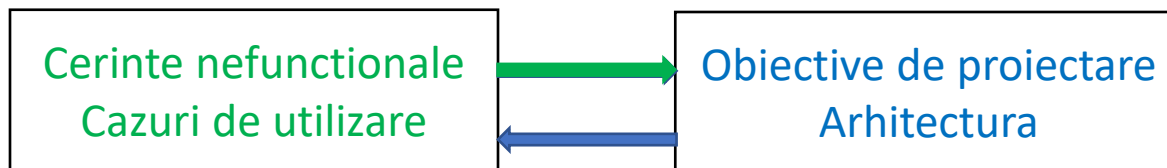
1. Au fost obiectivele proiectarii realizate?

2. Este modelul de proiectare arhitecturala:

- ✓ corect
- ✓ complet
- ✓ consistent
- ✓ realist
- ✓ usor de inteles

Verificarea proiectarii arhitecturale (2)

- ❖ Modelul de proiectare este corect daca modelul de analiza se poate mapa pe modelul de proiectare si invers:



- Poate fi mapat fiecare caz de utilizare pe un subsistem/set de subsisteme?
- Exista pentru fiecare subsistem un caz de utilizare sau o cerinta nefunctionala?
- Este fiecare cerinta nefunctionala adresata in modelul de proiectare?
- Poate fi mapat fiecare obiectiv de proiectare pe o cerinta nefunctionala?
- Exista pentru fiecare actor o politica de acces?
- Este fiecare politica de acces consistenta cu cerinta de securitate?

Verificarea proiectarii arhitecturale (3)

❖ **Modelul este complet daca toate cerintele au fost adresate in modelul de proiectare:**

- Exista functionalitati in cazurile de utilizare care nu se regasesc in subsistemele proiectate?
- Sunt prevazute conditiile limita?

❖ **Modelul este consistent daca nu exista contradictii:**

- Sunt prioritizate obiectivele de proiectare conflictuale?
- Exista obiective de proiectare in contradictie cu cerinte nefunctionale?
- Exista mai multe subsisteme sau clase cu acelasi nume?

❖ **Modelul este realist daca poate fi implementat**

❖ **Modelul este usor de inteles daca dezvoltatorii neimplicati in proiectarea de sistem înțeleg modelul.**

Lecturi suplimentare – proiectarea arhitecturală

1. Ian Sommerville, Software Engineering, Tenth Edition, Pearson Education Limited, 2015, ISBN 13: 978-1-292-09613-1
2. <https://www.slideshare.net/software-engineering-book/ch6-architectural-design>
3. <http://software-engineering-book.com/web/archpatterns/>
4. Mark Richards, Software Architecture Patterns, O'Reilly Media, Inc, 2015, ISBN: 9781491971437
5. Rick Kazman, Paul Clements, Len Bass, Software Architecture in Practice, Third Edition, Addison-Wesley Professional, 2012, ISBN: 9780132942799
6. MVC Tutorial for Beginners: What is, Architecture & Example: <https://www.guru99.com/mvc-tutorial.html>
7. https://www.uml-diagrams.org/examples/website-admin-use-case-diagrams-example.html#google_vignette