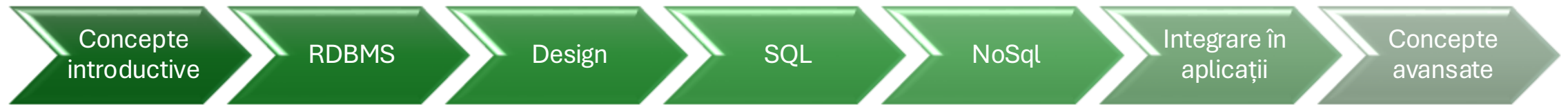


Baze de date

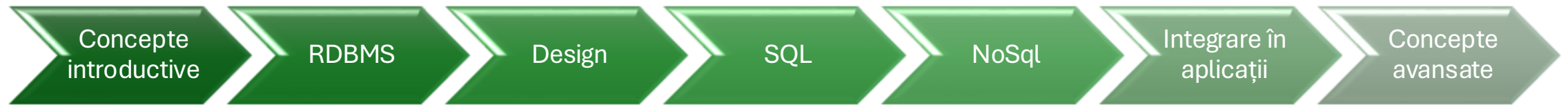
anul 1, semestrul 2

Curs 1

- Obiective.
- Informații organizatorice.
- Informații despre evaluare.
- Introducere design RDB.

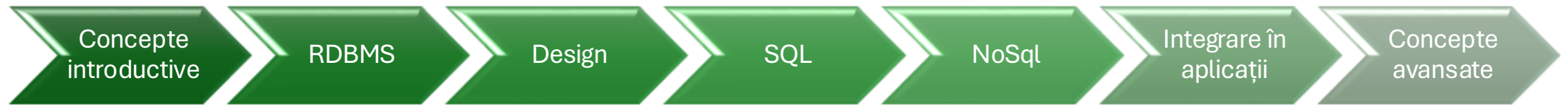


- **DBMS-concepte introductive**, cuvinte cheie.
- stocare, actualizare, regăsire date.
- suport pentru aplicații
- securitate: date protejate.
- persistență: mecanisme de *recovery*.
- concurență: acces simultan pentru citire și/sau actualizare.
- *Consistency*: cum sunt respectate regulile (*regulile de integritate*).

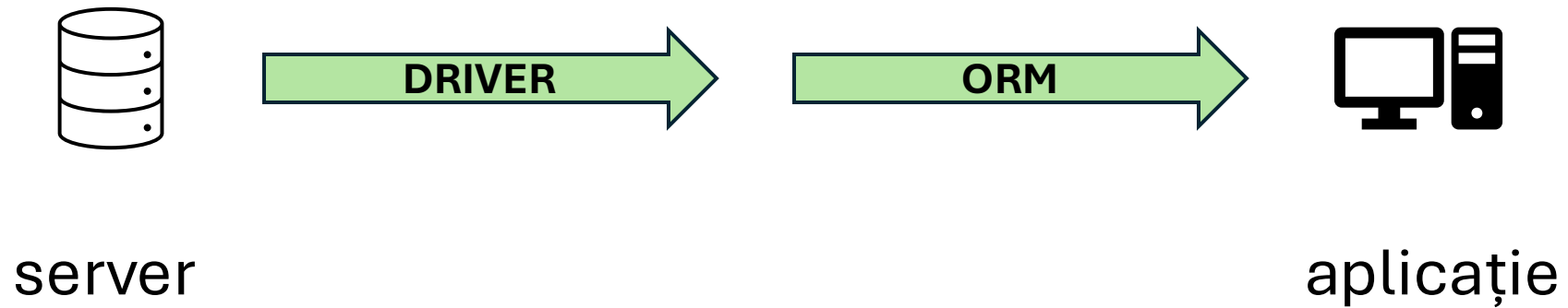


- **RDBMS**

- Datele sunt structurate matematic sub formă de *relații* i.e. mulțimi de *tupluri*, asupra cărora putem opera cu *operatori* specifici: join, reuniune, intersecție, division etc.
- **TABELE** cu LINII și COLOANE.
- **LINIE**: instanță.
- **COLOANĂ**: proprietate.



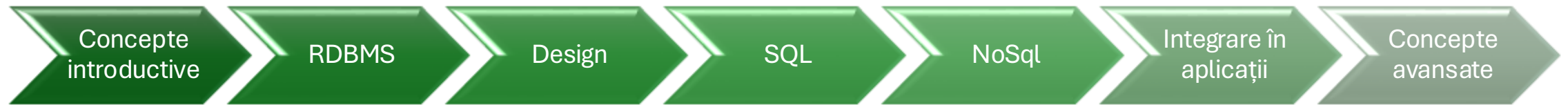
- **DB/RDBMS**





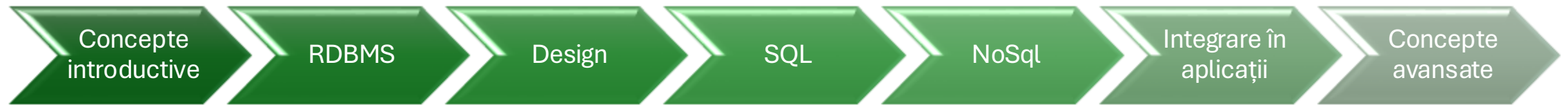
- **RDBMS**

- structură fixă .
- securitate.
- integritate.
- consistență.
- fără redundanțe.
- model ACID pentru tranzacții.

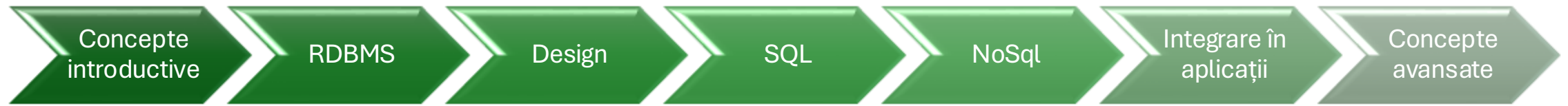


- **RDBMS**

- Limbaj declarativ standard pentru definirea, actualizare și manipularea datelor.
- **SQL**: LDD, LMD, LCD.
- Dicționar al datelor care cuprinde metadate și care poate fi consultat prin interogări SQL.
- Obiecte utilizate pentru optimizarea cererilor (index, view, cluster).
- Posibilitatea de a scala vertical, orizontal.
- Posibilitatea de a replica și partiționa date.

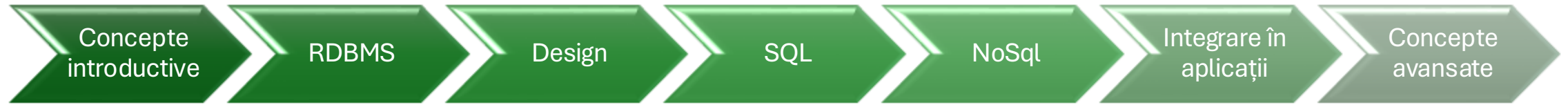


- **RDBMS** cum asigură eficiența și scalabilitatea?
- Obiecte utilizate pentru optimizarea cererilor (index, view, cluster, partiție).
- Posibilitatea de a **scala**
 - Vertical: mai multă putere de calcul pentru un server
 - Orizontal: mai multe servere.
- Posibilitatea de a
 - **Replica**: păstrăm mai multe copii ale datelor
 - **Partiționa**: distribuim datele în funcție de chei de partiționare.



- **Design**

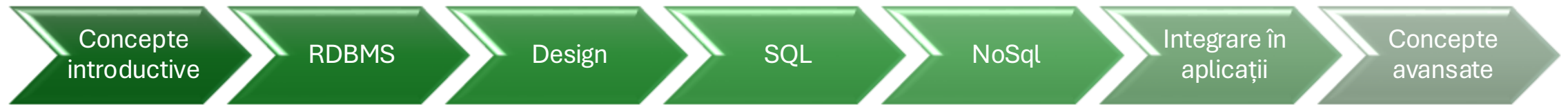
- Proiectăm o bază de date relațională la nivel conceptual pentru a putea servi o aplicație web sau mobile
 - hotel pentru animale de companie.
- Normalizăm sau proiectăm normalizat: Ne asigurăm că modelul nu va conține redundanțe care pot genera anomalii în operațiile de actualizare a datelor.
 - nu vrem să regăsim în aplicație pentru același oaspete și numele Gardfield și numele Grumpy.



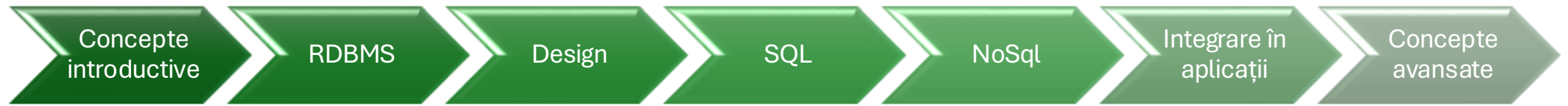
- **SQL**

- Implementăm modelul **SQL-LDD**:

- creăm **tabelele** într-o instanță de server Oracle
 - apartinator, oaspete, rezervare, spațiu etc.
- definim **constrângerile**.
 - nu vrem ca un spațiu să fie ocupat simultan de doi oaspeți
- definim alte obiecte pentru optimizarea performanței (răspuns rapid la cererile care vin din aplicație).
 - vrem ca pagina cu spații disponibile la o anumită dată să se încarce rapid.



- **NoSql**
- Denormalizare: Sunt anumite redundanțe necesare?
 - Putem păstra numele și informațiile despre Jon de două ori ca să găsim rapid informații despre stăpânul lui Gardfiel sau despre aparținătorul lui Odie.
- Este necesară o schemă flexibilă?
 - Ce vrem să prezentăm în oferta de rezervare (spațiu, hrană, asistență medicală)
- Cum putem scala
 - Cum răspunde aplicația dacă avem mai mulți clienți pentru spații care sunt situate în zone depărtate din țară?

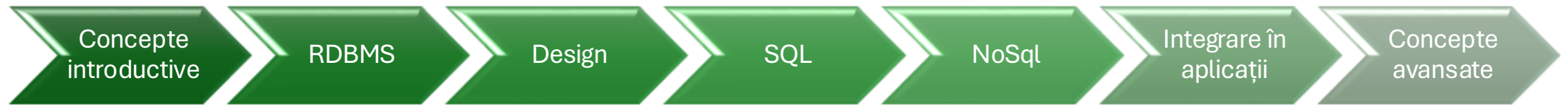


Relational

- **Vertical**/horizontal scalability
- Consistency model: ACID
- pre-defined schema
- SQL language
- Normalized data

NoSql

- Vertical/**Horizontal** scalability
- Consistency model: BASE
- Flexible schema
- No standard
- Collections, redundancy



- **Alte concepte**

- vector dbs

- transform large objects into vectors storing numeric values.
 - embedding.
 - clustering and similarities: calculate distances between vectors, llm context

- serverless architectures

- Lambda function care actualizează automat disponibilitatea în urma unei rezervări și care notifică automat clientul prin mail.

- graph Databases for relationships and recommendations.

*validare temă,
feedback, revizuire*

30 martie 2024, ora
23:59

- **Tema proiectului**
- **Diagrama ERD, Diagrama conceptuală**
- cerințe 1, 6, 7.

*validare cerințe obligatorii
revizuire complexitate*

11 mai 2024, ora
23:59

- Finalizare proiect, cerințe 1-13
- **obligatoriu 4 puncte**

pregătire prezentare

06 iunie 2024, ora
23:59

- Finalizare proiect, cerințe 14-17
- Revizuire pentru complexitate fără modificarea structurii validate la 1-13.
- **maxim 3 puncte**

sesiune, planificare
pe grupe

- **obligatoriu 2** întrebări design
- **obligatoriu 2** întrebări SQL
- Întrebări privind cerințele 14-17.
- Întrebări privind complexitatea proiectului.

teme/teste suplimentare curs și laborator **maxim 2 puncte**