



UNIVERSITÀ
DEGLI STUDI
DI BERGAMO

Programmazione Web

Linee guida per il
Progetto #1

DOCENTE
Paolo Fosci

EMAIL
paolo.fosci@unibg.it

TITOLARE DEL CORSO
Prof. Giuseppe Psaila

- Gruppi di 2 o 3 persone
- Form per iscrizione del gruppo:
<https://forms.office.com/e/DT1aHeJpBJ>
- In via eccezionale sono consentiti casi singoli (scrivetemi una mail)
- Il form rimane aperto fino a fine Aprile 2025
- I primi progetti saranno comunicati entro la metà di Aprile 2025



- Per il primo progetto dell'anno ad ogni gruppo verrà dato lo schema di un database differente.
 - Dovete crearlo sulla vostra piattaforma e popolarlo in maniera massiva (usate pure fonti esterne, o create dati sintetici con qualunque sistema riteniate utile – **l'importante è avere una certa numerosità e varietà nei dati**)
- Dovrete sviluppare un'applicazione per la quale vi verrà fornito un template di interfaccia (uno per gruppo)
- Ogni applicazione dovrà realizzare delle pagine per cercare dati per ognuna delle tabelle del database
- Laddove siano i dati siano linkabili, creare i suddetti link.
- Ogni applicazione dovrà realizzare tutte le operazioni di CRUD per una tabella particolare che vi verrà indicata.

1. Di seguito vi vengono dati i dettagli (contesto e specifica funzionale corredati di schema concettuale e logico) di ogni database di riferimento.
2. Creare ogni tabella che vedete nello **schema logico** e popolarla rispettando le relazioni tra le tabelle.
3. Quando create il database non importa se nel database impostate o meno le chiavi e/o i vincoli referenziali, questi li dovreste gestire in caso tramite l'applicazione
4. Le tabelle sulle quali creare una pagina di ricerca sono quelle **azzurre** negli schemi concettuali.

Elenco dei database di riferimento (nelle pagine seguenti la descrizione di ognuno di essi:

| # | Database | Riferimento |
|---|--------------------|-------------|
| 1 | Museo | Ex 1 |
| 2 | Telefoni | Ex 2 |
| 3 | Servizio Sanitario | Ex 3 |
| 4 | Ricettario | Ex 4 |
| 5 | Cinema | Ex 5 |
| 6 | Social Network | Ex 6 |

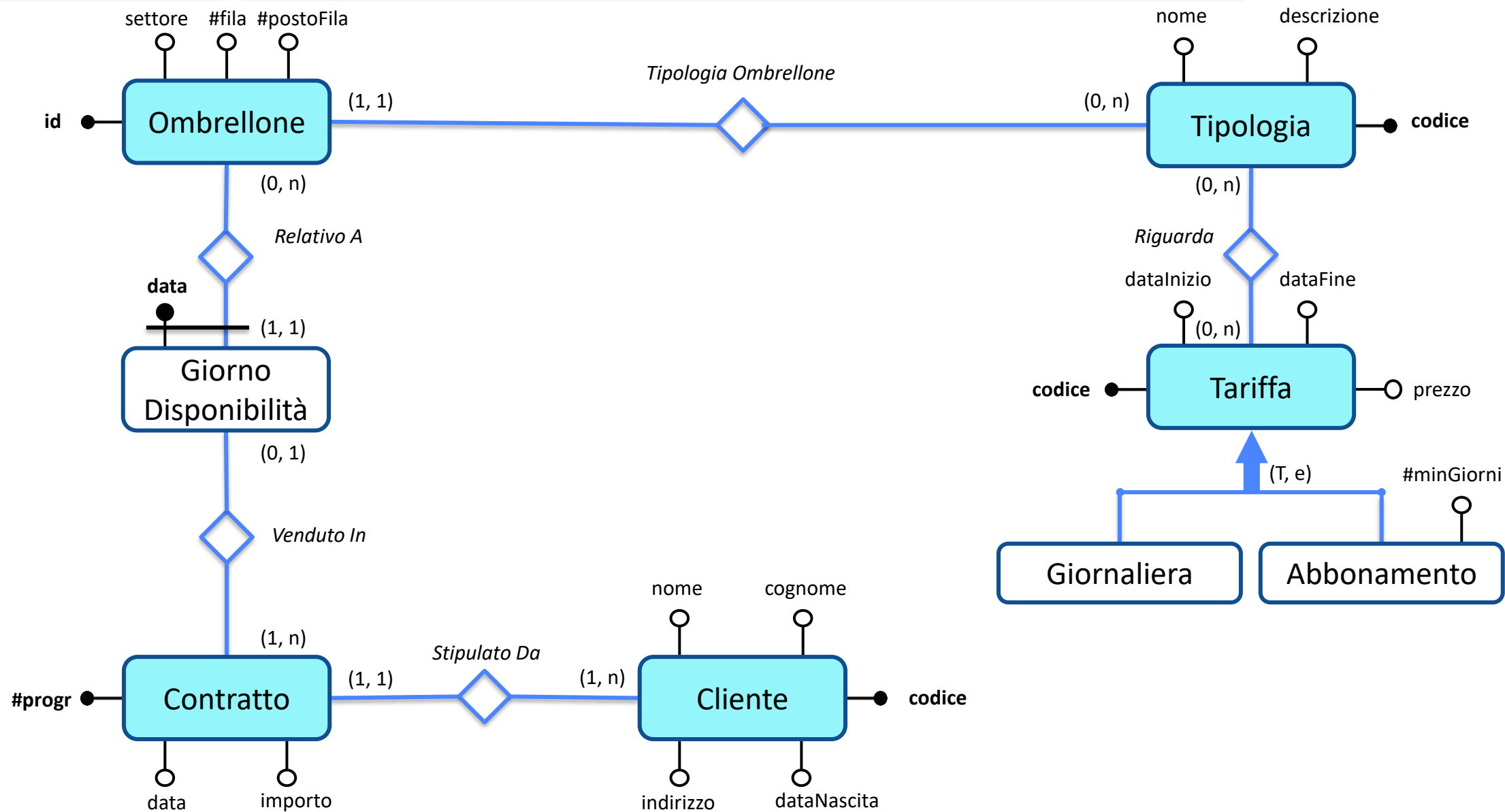
| # | Database | Riferimento |
|----|---------------------|---------------|
| 7 | Ombrelloni | Ex 7 |
| 8 | Autostrade | TE 08-06-2018 |
| 9 | Motorizzazione | TE 09-09-2019 |
| 10 | Distribuzione Acqua | TE 20-01-2020 |
| 11 | Quiz | TE 03-07-2020 |
| 12 | Palestra | TE 29-06-2021 |

Base di dati per la gestione del noleggio degli ombrelloni in una spiaggia attrezzata in cui si ha la necessità di gestire l'affitto degli ombrelloni ai clienti, in base al tipo di ombrellone e al periodo.

Ogni ombrellone è identificato da un identificatore numerico, ed è caratterizzato dal settore della spiaggia, dal numero di fila e dal numero d'ordine all'interno della fila. Gli ombrelloni sono associati ad una tipologia, dove ogni tipologia è identificata da un codice ed è caratterizzata da un nome e dalla descrizione (testuale) degli accessori in dotazione agli ombrelloni di quella tipologia (per esempio, sdraio, lettino, ecc.). Per ogni tipologia, si ha un insieme di tariffe associate: le tariffe indicano quale prezzo applicare a seconda del periodo e del tipo di affitto che viene scelto dal cliente. Pertanto, una tariffa è identificata da un codice ed è caratterizzata dal periodo di validità della tariffa, nonché dal prezzo; inoltre, le tariffe vengono suddivise in giornaliera (che valgono per un affitto di un solo giorno) o in abbonamento e per queste ultime si vuole sapere il numero minimo di giorni per far decorrere l'abbonamento.

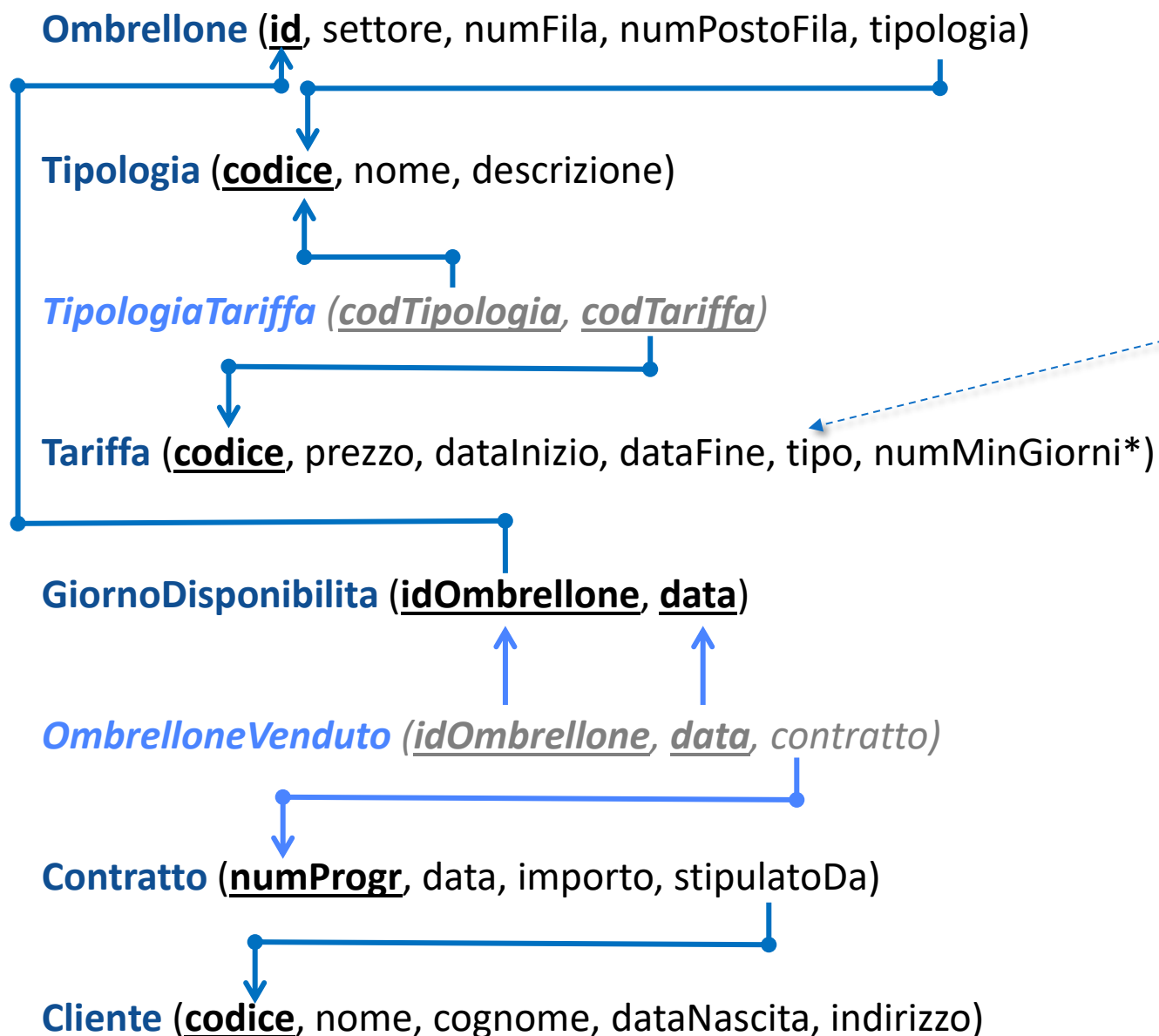
Per poter affittare gli ombrelloni senza correre il rischio di affittare lo stesso ombrellone a due clienti contemporaneamente, occorre predisporre, per ciascun ombrellone, un insieme di giorni di disponibilità: ogni giorno di disponibilità è identificato univocamente dalla data rispetto all'ombrellone di riferimento (ovviamente, possono esserci giorni di disponibilità con la stessa data ma per ombrelloni diversi).

Per finire, l'ufficio vendite effettua un contratto di affitto con un cliente; il contratto è identificato da un numero progressivo ed è caratterizzato dalla data, dall'importo complessivo e dai giorni di disponibilità degli ombrelloni affittati con quel contratto (ad un giorno di disponibilità può essere associato al più un contratto).



Legenda:

- **sottolineato** : attributo chiave
- * : attributo annullabile



Constraint su Tariffa:

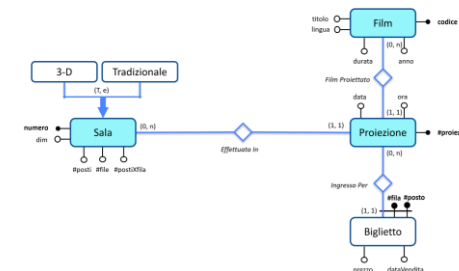
(tipo='Giornaliera' AND numMinGiorni IS NULL)

OR

(tipo='Abbonamento' AND numMinGiorni IS NOT NULL)

**QUESTO E' UN ESEMPIO DI STRATEGIA PER POPOLARE IL DATABASE RELATIVO AL CASO DEL CINEMA:
PER I CASI SPECIFICI DEI VOSTRI PROGETTI FATE IPOTESI VEROSIMILI
PER AVERE DATI CONSISTENTI E VARIEGATI**

- Cerchiamo una lista di film (magari qualche centinaio)
- Possiamo immaginare che il cinema abbia un certo numero di sale (es.: da 4 a 15?)
 - ognuna con un certo numero di posti (es: da 20 a 400) – gli altri dati inventateli
- Per le proiezioni immaginiamo un periodo di gestione del cinema (es: 3, 4 anni, tipo dal 2021 al 2025)
- Ipotizziamo che circa l'80% dei film venga proiettato. Poi per ogni film proiettato consideriamo
 - prima proiezione in un giorno a caso del nostro periodo di riferimento (spalmiamo i film nel periodo)
 - successivamente il film viene proiettato secondo una gaussiana da 1 a 21 giorni o qualcosa di simile
- Per ogni giorno di proiezione possiamo immaginare di aver un numero variabile di proiezioni in diversi orari (da 1 a 4 o 5) – in una certa sala scelta a caso
- **Volendo fare le cose bene,**
controllare di non avere due proiezioni differenti nella stessa sala con stesso giorno e orario
- Per ogni proiezione, secondo una gaussiana, possiamo considerare un numero di biglietti in base alla capacità della sala



Le query di ricerca su una tabella devono prevedere diversi criteri (dipende dalla tabella). In generale:

- se una tabella A è legata da una relazione (0:1) o (1:1) ad un'altra tabella B sarebbe il caso di mostrare insieme alla riga di A i dati caratteristici di B (oltre all'ID). Es.: nel **DB1**, insieme ai dati di un'Opera mostrare il nome dell'Autore
- se una tabella A è legata da una reazione (0:n) o (1:n) ad un'altra tabella B allora mostrate il numero di entità di B legate ad ogni riga di A. Es.: nel **DB1**, insieme ai dati di un Autore mostrare il numero di Opere

Fate le vostre considerazioni caso per caso

Di seguito sono elencati i template di riferimento per le interfacce.

I template sono solo di riferimento solo nel senso del posizionamento:

dimensioni, proprietà e colori sono a scelta vostra in base alle esigenze della vostra applicazione.

Se il template si adatta poco alla vostra applicazione (es: il form di ricerca richiede più spazio) fate le vostre valutazioni per eventuali modifiche.

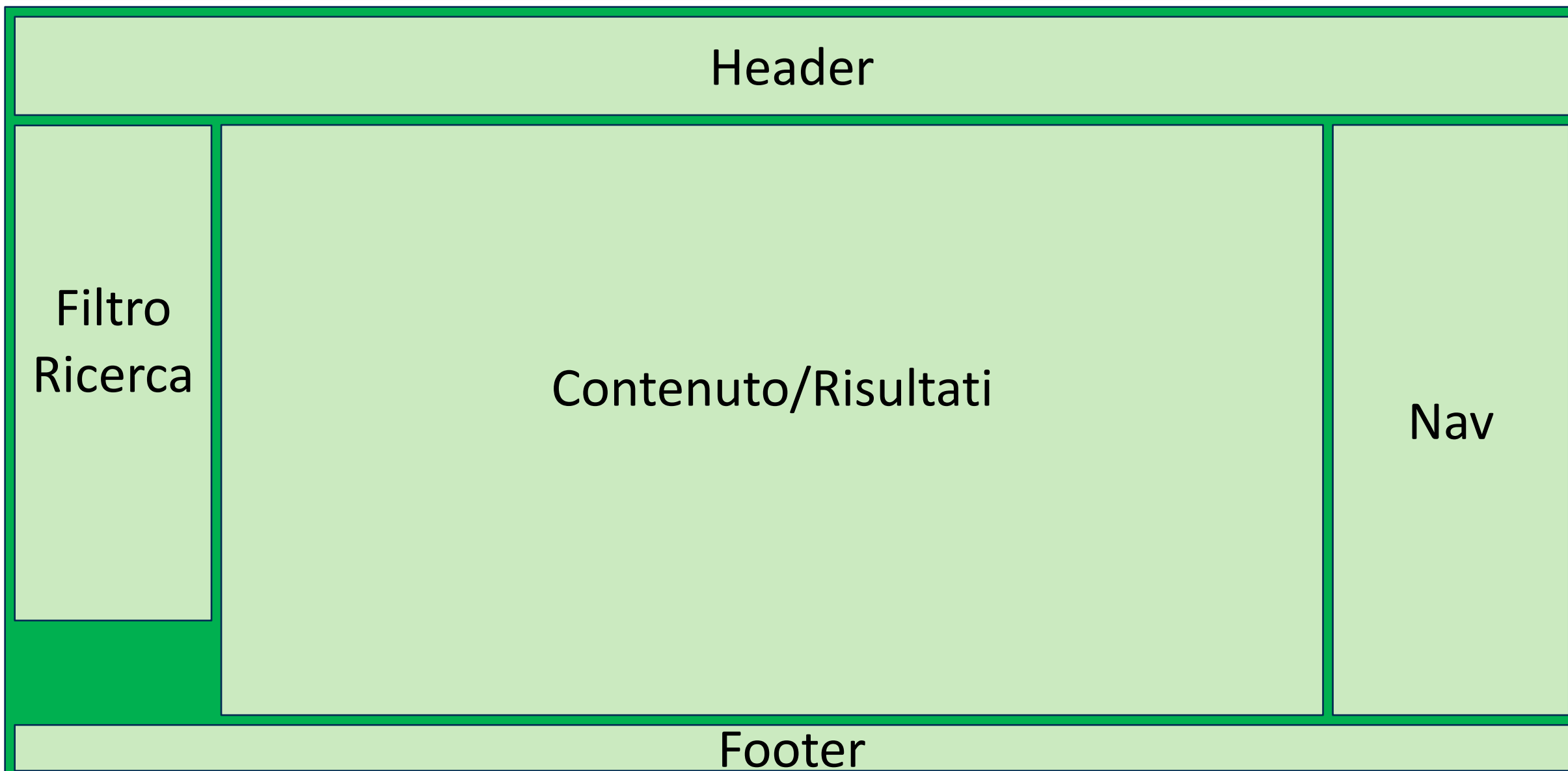
Le interfacce devono essere il più robuste possibile a cambiamenti di dimensione del browser

Vi verrà assegnata una palette di riferimento,

il che NON vuol dire che gli sfondi devono avere tutti una stessa tonalità!

Significa che dovete avere uno stile (per sfondi, caratteri o altri elementi tipo le icone) che richiami il colore della palette. Non vuol dire che tutti gli elementi devono avere certe tonalità, potete anche usare dei contrasti, anche forti, purché tutto sia coerente, e si capisca comunque qual è la tonalità dominante.

Potete usare anche delle immagini per gli sfondi e tutti gli elementi creativi ed effetti che ritenete di dover usare!



- Flaticons
<https://www.flaticon.com/>
- PNG Arts
<https://www.pngarts.com/it/>
- Pixabay
<https://pixabay.com/>
- Meat Ball Studios
<https://metaballstudios.creator-spring.com/>

Acronimo per tutte quelle operazioni sui dati relative a:

- Create
- Read
- Update
- Delete

Cosa si aspetta l'utente medio? – E quello dummy?
Considerate che ci sono casi in cui alcune operazioni sono consentite solo a gruppi ristretti di utenti...

Se l'utente deve inserire dei nuovi dati in database:

- che strumenti dobbiamo mettergli a disposizione?
- che messaggistica dobbiamo fornirgli?
- quali dati deve poter inserire?
 - gli ID deve fornirli oppure no?
 - come gestire eventuali chiavi esterne?

Dobbiamo considerare due aspetti:

1. Che dati dobbiamo fornire all'utente?
 - L'utente deve essere conscio della struttura interna dei dati?
 - I dati mostrati all'utente debbono mantenere lo stesso formato interno?
2. Quali strumenti mettiamo a disposizione all'utente
 - Per cercare e filtrare i dati?
 - Le ricerche debbono essere stringenti oppure lasche?
 - Per ordinare i dati?

Se l'utente deve modificare dei dati in un database:

- che strumenti dobbiamo mettergli a disposizione?
 - Ci sono funzionalità che possiamo sfruttare?
- che messaggistica dobbiamo fornirgli?
- quali dati deve poter modificare?
 - Ci sono dati che debbono essere protetti più di altri?

Se l'utente deve cancellare dei dati in un database:

- che strumenti dobbiamo mettergli a disposizione?
- che messaggistica dobbiamo fornirgli?
- quali dati deve poter cancellare?
- Che fare se ci sono dipendenze?
 - Gestiamo la cancellazione a cascata?

Who, When, What, Where, How:

- A chi devo comunicare?
- Quando devo avvertire l'utente?
- Cosa debbo comunicargli?
- Dove?
- Come lo comunico?

- Potete usare solo le tecnologie presentate finora: (HTML, CSS, JS, PHP, JQUERY, AJAX)
- Mail con [PW25] nell'oggetto insieme al nome del gruppo e del progetto
 - Tutti i componenti in cc
 - Link al sito
 - Zip file con il codice
 - Excel o Ods usato/i per popolare il database
 - Una paginetta di documentazione/spiegazione...

Usate un po' di fantasia, buon senso
e buon lavoro!

... e se avete problemi particolari scrivetemi una mail
(con il prefisso PW25)