***Innovative Buzzi***

|  |  |  |
| --- | --- | --- |
| Nome | Classe | Anno scolastico |
| Federico Giacumbo | 5°Q | 2019-2020 |
| Davide Hu |
| Claudiu-Cassian Ferencz |

|  |  |  |  |
| --- | --- | --- | --- |
| Versione | Data | Modifiche effettuate | Autore |
| 1.0 | 08/04/2020 | Prima versione | Federico Giacumbo |

**Descrizione dell’esigenza e piano progetto**

1. **Scopo del progetto**

Adattare la gestione della sala stampa del Buzzi con l’ausilio di nuove tecnologie.

1. **Perimetro ed ambiti di intervento**

La realizzazione di questo progetto aumenta la produttività della sala stampa riducendo al minimo i tempi di attesa.

1. **Vincoli**

Consegnare il progetto entro il 30 aprile 2020

1. **Deliverables attesi**

Struttura database realizzato, Applicazione web tramite browser.

1. **Team di progetto / gruppo di lavoro**

Federico Giacumbo = P. M gruppo di progetto

Claudiu-Cassian Ferencz = Membro del gruppo di progetto

Davide Hu = Membro del gruppo di progetto



1. **Piano delle attività**

****

****

1. **Costi di progetto**

* 3 Computer per poter lavorare contemporaneamente
* Web server per ospitare il sito web
* Database

**Analisi dei requisiti**

1. **Situazione as-is**

Nell’attuale situazione per richiedere una stampa bisogna recarsi fisicamente alla sala stampa compilando l’apposito modulo con classe, richiedente(nome cognome), numero copie necessarie, a pagamento o gratuito( per l’insegnante è gratuito) ed eventuali note. Dunque, si dovrà aspettare il proprio turno davanti alla sala stampa e attendere che vengano fatte le copie.

Diagramma di flusso interfunzionale:



1. **Situazione to-be**

Dopo la realizzazione di questo sito web sarà possibile mandare il file da stampare a distanza, riducendo tempo inutile in coda. Per usufruire di tale servizio telematico, sarà obbligatorio creare un account altrimenti sarà comunque possibile stampare recandosi fisicamente in sala stampa. Il cliente si troverà di fronte ad un computer nel quale accederà con le credenziali di un account fornitogli dalla segreteria e immettere il suo nominativo.

Digramma dei casi d’uso:



1. **Analisi degli impatti organizzativi**

Per il corretto funzionamento del sito web, sarà necessaria la presenza di 2 operatori: uno che si occuperà delle richieste di stampa e sarà situato davanti al computer mentre l'altro svolgerà l'azione di stampa. È consigliato partecipare a dei corsi di durata di circa 2-3 ore, organizzati dal team di sviluppo per comprendere a pieno il funzionamento del sito che sarà completo al suo rilascio.

1. **Requisiti di dominio**

Per accedere al servizio di stampa a distanza è necessario essere in possesso di un account e quindi effettuare il login.

**Analisi e progettazione della base di dati**

**Progettazione concettuale**

Dizionario delle entità

|  |  |  |  |
| --- | --- | --- | --- |
| Termine | Sinonimo | Descrizione | Collegamento |
| Utente | // | Utente che fa la prenotazione per la stampa online | Stampa, Prenotazione |
| Professore | // | Sottoentità di “Utente”, professore che fa la prenotazione online |  |
| Studente | // | Sottoentità di “Utente”, studente che fa la prenotazione online |  |
| Prenotazione | // | Per prenotazione si intende la richiesta di stampa che può essere fatta a distanza mandando il file o direttamente alla sala stampa consegnando il materiale | Utente, File, Operatore |
| File | // | Contenuto che deve essere stampato presente all’interno della prenotazione | Prenotazione |
| Operatore | // | Sotto entità di utente, Membro del personale che si occupa di gestire le prenotazioni e stampare | Prenotazione, Stampa |
| Stampa | // | Fotocopia del materiale (online/fisico) da stampare | Operatore, Formato, Utente, Studente |
| Formato | // | Formato della stampa che ne determina il costo | Stampa |
| Prodotto | // | Materiale che è possibile comprare oltre alle fotocopie | Utente |

Dizionario dei dati

|  |  |  |  |
| --- | --- | --- | --- |
| Entità | Attributo | Tipo | Descrizione |
| Utente | codiceFiscale | char(16) | Identificativo dell’utente (Chiave primaria) |
| email | varchar(64) | email con la quale l’utente verrà identificato |
| password | varchar(256) | password dell’utente criptata. |
| nome | varchar(32) | nome dell’utente |
| cognome | varchar(32) | cognome dell’utente |
| dataNascita | date | data di nascita dell’utente |
| Indirizzo | varchar(32) | Attributo composto formato da via,civico,cap |
| Professore | // | // | // |
| Studente | classe | TINYINT | classe dello studente |
| sezione | varchar(3) | sezione della classe dello studente |
| Operatore | // | // | // |
| Prenotazione | idPrenotazione | int | pk int auto increment |
| quantità | int | numero di copie richieste da stampare |
| dataPrenotazione | date | data in cui si è effettuata la prenotazione |
| oraPrenotazione | time | ora in cui si è effettuata la prenotazione |
| stampata | ENUM | indica se la prenotazione è stata stampata o meno(“si” significa che è stata stampata,”no” significa che non è stata stampata) |
| File | idFile | int | pk int auto increment |
| nomeFile | varchar(128) | nome del file |
| dimensione | int | dimensione del file |
| Stampa | idStampa | int | int autoincrement pk |
| dataStampa | date | data in cui la stampa ha avuto luogo |
| oraStampa | time | ora in cui la stampa ha avuto luogo |
| Formato | costoStampa | decimal(3,2) | costo della stampa |
| tipo | varchar(16) | tipo del formato (chiave primaria) |
| Prodotto | idProdotto | int | int autoincrement pk |
| nomeProdotto | varchar(64) | nome del prodotto venduto in sala stampa |
| costoProdotto | decimal(4,2) | prezzo del prodotto venduto in sala stampa |

Dizionario delle associazioni

|  |  |  |  |
| --- | --- | --- | --- |
| Associazione | Entità coinvolte | Descrizione | Attributi |
| Fa | Utente, prenotazione | L’utente fa la prenotazione per stampare | // |
| Ritira | Utente, stampa | L’utente ritira il file stampato | dataRitiro, oraRitiro |
| Contiene | Prenotazione, file | La prenotazione contiene il file da stampare | // |
| Gestita | Prenotazione, operatore | L’operatore gestisce le prenotazioni(inclusa la stampa). | // |
| Ha | Stampa, formato | La stampa ha un formato | // |
| Paga | Studente, stampa | Solo lo studente paga la stampa | // |
| Acquista | Utente, Prodotto | l’utente può acquistare un prodotto | // |

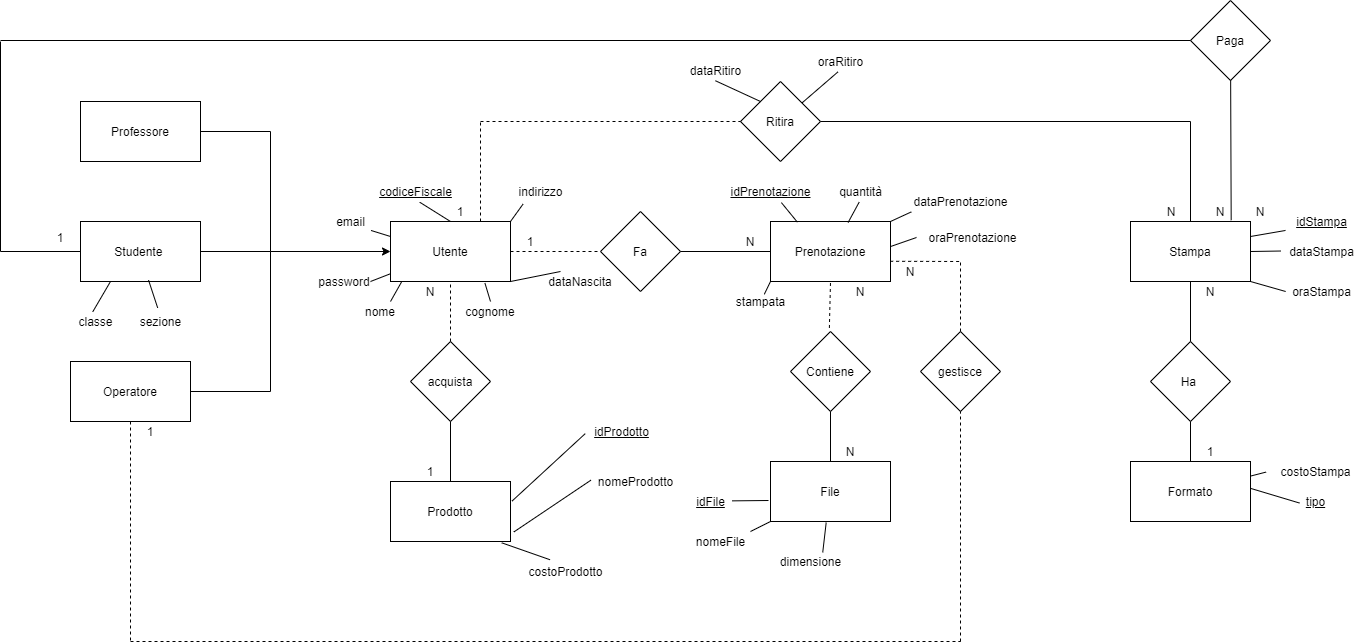
Note

Nella tabella delle prenotazioni è presente come chiave esterna quella dell’utente.L’utente può essere anche un operatore. Nel caso quindi “Gestirà” la richiesta.

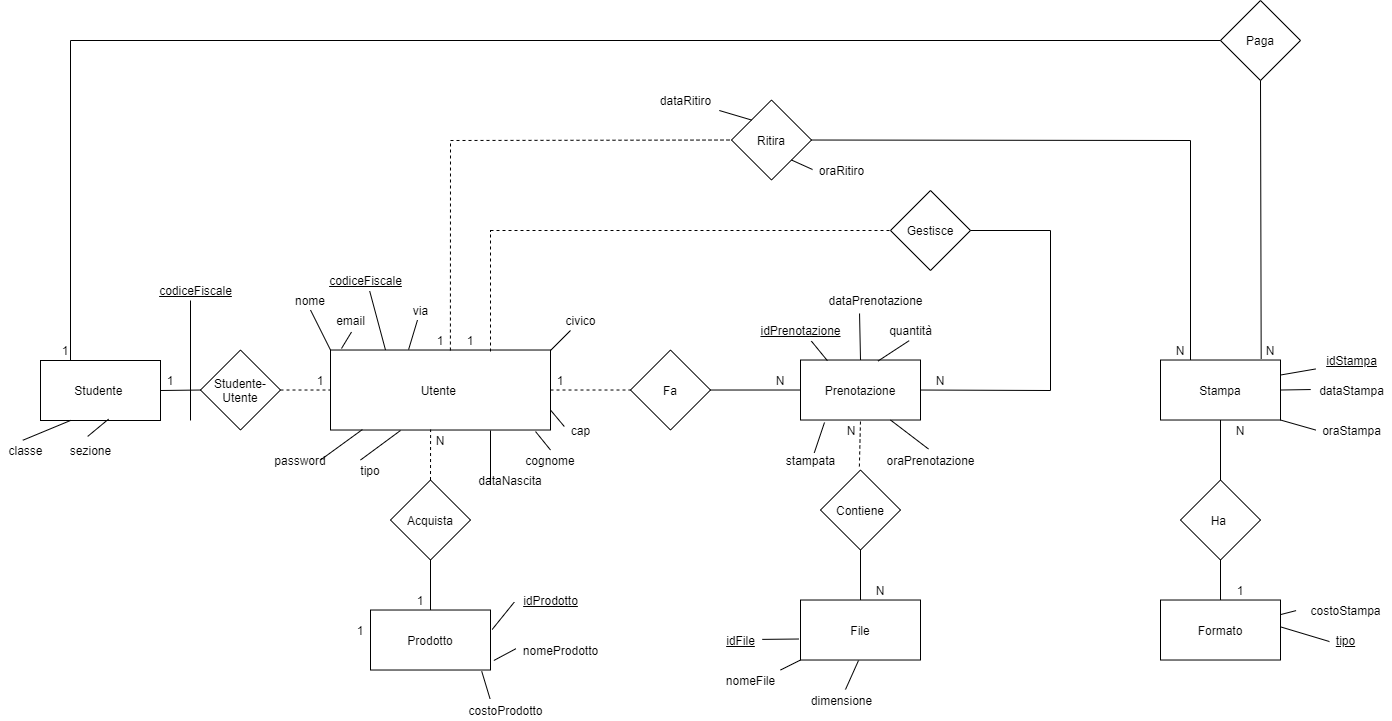
**Ipotesi aggiuntive**

* Gli utenti (professori e studenti) sono stati già registrati in quanto si usano le credenziali del vecchio registro. Se non si usassero vecchi dati, perchè eliminati per il nuovo registro, verranno creati dei dati nuovi a ciascun studente e professore così da consentire l’utilizzo del servizio ai soli membri interni della scuola.
* Gli utenti (professori e studenti) sono registrati di default. In questo modo è possibile salvare e tenere traccia dei prodotti acquistati.

**Modello ER**

****

**Ristrutturazione modello ER**

****

**Progettazione Logica**

**Schema logico:**

Utente(codicefiscale,nome, cognome,email,password,tipo,dataNascita,civico,cap, ,idProdotto\*)

Studente(codiceFiscale\*,classe,sezione)

Prodotto(idProdotto,nomeProdotto,costoProdotto)

Stampa(idStampa,dataStampa,oraStampa,codiceFiscaleUtente\*,dataRitiro,oraRitiro,tipoFormato\*, codiceFiscaleOperatore\*)

Formato(tipo,costoStampa)

Prenotazione(idPrenotazione,dataPrenotazione,quantità,oraPrenotazione,stampata,codiceFiscale\*)

Contiene(idFile\*,idPrenotazione\*)

File(idFile,nomeFile,dimensione)

**Progettazione fisica**

**Query**

Create database innovativeBuzzi;

Use innovativeBuzzi;

Create table Prodotto(

idProdotto int auto\_increment primary key,

nomeProdotto varchar(64),

costoProdotto decimal(4,2)

);

create table Utente(

codiceFiscale char(16) primary key,

nome varchar(32),

cognome varchar(32),

password varchar(256),

email varchar(64),

tipo varchar(64),

dataNascita DATE,

civico int,

cap int,

idProdotto int,

FOREIGN KEY(idProdotto) REFERENCES Prodotto(idProdotto)

ON UPDATE CASCADE

);

Create table Studente(

codiceFiscale char(16),

classe char(3),

sezione TINYINT,

FOREIGN KEY(codiceFiscale) REFERENCES Utente(codiceFiscale)

ON UPDATE CASCADE,

Primary key(codiceFiscale)

);

Create table Formato(

Tipo varchar(16) primary key,

costoStampa decimal(3,2)

);

Create table Stampa(

idStampa int auto\_increment primary key,

dataStampa DATE,

oraStampa TIME,

codiceFiscaleUtente char(16),

dataRitiro DATE,

oraRitiro TIME,

tipoFormato varchar(16),

FOREIGN KEY(codiceFiscaleUtente) REFERENCES utente(codiceFiscale),

FOREIGN KEY(tipoFormato) REFERENCES formato(tipo)

ON UPDATE CASCADE

);

Create table Prenotazione(

idPrenotazione int auto\_increment primary key,

dataPrenotazione DATE,

oraPrenotazione TIME,

quantità int,

stampata ENUM("si","no") default "no",

codiceFiscaleCliente char(16),

FOREIGN KEY(codiceFiscaleCliente) REFERENCES utente(codiceFiscale)

ON UPDATE CASCADE

);

Create table File(

idFile int auto\_increment primary key,

nomeFile varchar(128),

dimensione int

);

Create table Contiene(

idFile int,

idPrenotazione int,

FOREIGN KEY(idFile) REFERENCES File(idfile),

FOREIGN KEY(idPrenotazione) REFERENCES Prenotazione(idPrenotazione)

ON UPDATE CASCADE,

Primary key(idFile,idPrenotazione)

);

**Progettazione dell’applicazione**

**Allegati**