**UNIVERSITATEA ROMÂNO-AMERICANĂ**

**FACULTATEA DE INFORMATICĂ MANAGERIALĂ**



**LUCRARE DE LICENȚĂ**

Coordonator științific: Absolvent:

Lect.univ.dr. POP DRAGOȘ PAUL Burcea Claudia Elena

Grupa 635AS

București, 2025

**FACULTATEA DE INFORMATICĂ MANAGERIALĂ**



**Dezvoltarea aplicației “Student Hub”**

Coordonator științific: Absolvent:

Lect.univ.dr. POP DRAGOȘ PAUL Burcea Claudia Elena

Grupa 635AS

București, 2025

# **Introducere**

Nevoia de digitalizare este în continuă creștere, mai ales in mediul universitar, în special în rândul studenților care au crescut cu tehnologie și doresc ca instrumentele digitale sa ii ajute in fiecare aspect al educației lor. Soluțiile digitale, cum ar fi notitele din cloud, documentele colaborative, testele online și înregistrările de curs, nu mai sunt optionale, ci sunt acum considerate necesități. Odată cu comunicarea la distanță și mediile de învățare hibride care devin obișnuite, această tendință a devenit mai rapidă în ultimii ani. În plus față de digitalizarea instrumentelor tradiționale, studentii caută în mod activ platforme care să simplifice, să raționalizeze și să sincronizeze obligațiile academice.

Studentii folosesc o varietate de instrumente digitale si non-digitale pentru a-și organiza activitatea academică, precum servicii de stocare în cloud, cum ar fi Dropbox și Google Drive, aplicații pentru luare a notitelor virtuale precum Notion și Evernote, scrise fizic și aplicatii de time management precum Google Calendar sau Outlook. Deși fiecare dintre aceste instrumente are calități utile, ele nu sunt adaptate cerințelor unice ale vieții academice; acestea sunt concepute pentru uz general. Studentii sunt forțați să schimbe în mod continuu aplicațiile, sa duplice informații și sa le reorganizeze constant, ceea ce rezulta la timp pierdut în locul studiului ca urmare a acestei fragmentări. Prin urmare, o soluție unificată care combină toate nevoile de baza ale studentilor (scheduling, organizarea personala a orarului, monitorizarea deadline-urilor și planificarea examenelor) într-o manieră clară și coerentă este evident necesară.

Natura dispersată a conținutului academic este o altă problemă cu care se confruntă studenții, în plus față de utilizarea a numeroase aplicații cu scop general. Programele pot fi afișate într-un planificator fizic sau afisate pe platforma facultatii, temele pot fi discutate într-un grup WhatsApp, datele pot fi distribuite între un disc local, atașamente de e-mail și foldere cloud, iar actualizările cursului pot fi postate pe o platformă universitară. Studentii trebuie să asambleze în mod continuu contextul din mai multe surse, ca urmare a acestei distribuții dezorganizate, creând o experiență de învățare disjuncta. Există o șansă mai mare de a pierde termene-limită cruciale, de a pierde efortul sau de a pierde accesul la resurse cruciale atunci când nu există un sistem uniform în vigoare. În plus, creează o prejudecată personală în modul în care studentii își gestionează timpul, răspunzând frecvent la canalul care pare a fi cel mai urgent, mai degrabă decât cel mai important. Ca urmare, pe langa ineficiență, apare și un sentiment persistent de stres care poate afecta sănătatea mintală și performanța academică a studentilor.

În 2025, burnout-ul academic este mai frecvent ca niciodata și apare ca o problemă critică în rândul studenților. Cu orele de curs, termene limită, locurile de muncă și o stare perpetuă de disponibilitate, studentii se afla frecvent într-un ciclu stresant. Acest lucru se datorează în principal faptului că locurile de munca și instituțiile de învățământ au așteptări ridicate, fiecare anticipând dedicarea completă fără a lua în considerare cerințele celeilalte parti, ceea ce este în întregime rezonabil, dar greu de gestionat pentru un student. Această oboseală este agravată doar de neclaritatea liniilor care separă viața personală, intelectuală și digitală. Stresul cronic si epuizarea sunt amplificate de notificările diverselor platforme, presiunea de a fi productiv și lipsa timpului liber. Burnout este acum un eveniment normal, mai degrabă decât o boală rară, în special pentru studenții care nu dispun de resursele necesare pentru a-și gestiona în mod corespunzător timpul, materialele și încărcătura mentală. În prezent, soluțiile digitale inteligente care susțin echilibrul, claritatea și organizarea nu sunt doar practice, ci și necesare pentru sustenabilitatea academică si starea de bine a studentilor.

Aplicatia ce urmeaza a fi dezvoltata isi propune sa rezolve 2 mari probleme ale mediului academic universitar: infromatie ampla si fragmentata si burnout-ul academic si de a crea un mediu care sa sporeasca productivitatea. Se vor folosii tehnologii moderne precum React, Tailwind CSS, Shadcn UI și Elysia pentru backend, în timp ce baza de date va fi gestionată cu PostgreSQL și Drizzle ORM.

# **1. Studiul si analiza sistemului existent**

## 1.1 Modalitati actuale de organizare academica

Organizarea academică este o tehnică sistematică pe care elevii o folosesc pentru a-și gestiona îndatoririle academice, timpul și resursele pentru a-și îndeplini obiectivele educaționale în învățământul superior. Aceasta acoperă gestionarea claselor, sarcinilor, proiectelor și obiceiurilor de studiu. Organizarea academică eficientă este vitală pentru succesul academic al studenților, deoarece le permite să integreze îndatoririle academice cu activitățile extrașcolare și viața personală, reducând în cele din urmă stresul și îmbunătățind rezultatele. (Merkel, 2024)

Aditional orelor de curs din cardul facultatii, este necesara stabilirea si mentinerea unui program de studiu, intr-un interval al zilei specific. Facut corect si in mod repetat, acest lucru ajuta studentii sa fie mai organizati si transforma invatatul intr-o rutina zilnica. (Merkel, 2024).

Studenții utilizează instrumente digitale precum Google Calendar, Notion, Todoist sau aplicații de tip planner pentru a-și urmări temele, proiectele și examenele. Utilizarea acestor instrumente poate reduce procrastinarea și crește motivația (Bolden-Barrett, 2019). Totuși, cercetările arată că listele de sarcini foarte încărcate pot avea efect invers, crescând stresul și scăzând productivitatea dacă nu sunt gestionate corect.

Majoritatea studenților se bazează pe calendare digitale integrate (ex. Google Calendar, Outlook) pentru a nota termene limită, sesiuni de examene și alte date relevante. Acestea permit configurarea de memento-uri și sincronizarea cu alte aplicații educaționale. Planificarea vizuală s-a dovedit eficientă în creșterea ratei de îndeplinire a sarcinilor academice (Buehler et al., 2010)

Un aspect adesea neglijat în organizarea academică este planificarea timpului liber. Studiile arată că o bună organizare presupune echilibrarea muncii cu timpul de odihnă și recreere, ceea ce reduce burnout-ul și îmbunătățește performanța cognitivă (Salzer, 2012). Studenții care își notează din timp perioadele de vacanță sau pauzele planificate raportează o satisfacție mai mare și mai puține episoade de epuizare.

Materialele de curs (fișiere PDF, notițe, prezentări) sunt organizate, de regulă, în foldere digitale pe platforme cloud precum Google Drive, OneDrive sau Dropbox. Organizarea clară a acestor resurse, folosind denumiri standardizate și foldere tematice, contribuie la eficiența învățării și la reducerea timpului pierdut căutând fișiere (Saplacan et al., 2019). Cu toate acestea, fragmentarea între mai multe platforme poate crea confuzie și duce la pierderea sau dublarea materialelor.

Un element fundamental al organizării academice este **urmărirea termenelor limită** pentru teme, proiecte, teste și examene. Majoritatea studenților utilizează calendare digitale (ex. Google Calendar, Microsoft Outlook) sau planificatoare fizice structurate pe săptămâni, cu spații pentru priorități, sarcini zilnice și notițe. Planificarea vizuală a sarcinilor în timp ajută la reducerea fenomenului de *planning fallacy* – tendința de a subestima durata necesară finalizării unei activități (Buehler & Griffin, 2018). De asemenea, utilizarea memento-urilor automate în aplicații digitale s-a dovedit eficientă în creșterea ratei de îndeplinire a sarcinilor și reducerea procrastinării (Steel, 2007).

Prin înregistrarea proactivă a termenelor importante într-un sistem clar, studenții își pot structura mai bine săptămânile academice, pot evita acumularea de sarcini și pot reduce stresul asociat cu termenele iminente.

Organizarea academică nu implică doar gestionarea sarcinilor, ci și **planificarea pauzelor**. Studiile din domeniul sănătății mintale și educației subliniază importanța echilibrului dintre muncă și odihnă. Planificarea perioadelor de vacanță sau a weekend-urilor libere contribuie semnificativ la menținerea motivației și prevenirea sindromului de burnout (Salzer, 2012).

Instrumentele digitale oferă opțiunea de a marca perioadele de pauză în același calendar cu sarcinile academice, permițând o **viziune holistică** asupra echilibrului dintre muncă și relaxare. Studenții care își rezervă conștient timp pentru recuperare cognitivă și socială raportează o satisfacție mai mare în viața universitară și performanțe stabile în perioadele solicitante (Kuhlthau, 2021).

## 1.2 Instrumente digitale utilizate de studenți – avantaje și limitări

În contextul învățământului superior, studenții folosesc o varietate de aplicații și platforme digitale pentru a-și susține procesele de organizare, colaborare și învățare. Aceste instrumente, deși utile, prezintă atât beneficii evidente, cât și limite structurale care le pot afecta eficiența în utilizarea zilnică.

**Microsoft Teams**

* Avantaje: Microsoft Teams oferă un ecosistem integrat pentru colaborare academică – chat, videoconferințe, stocare de fișiere și calendar sincronizat cu Outlook. Este adesea folosit ca platformă centrală în universitățile care utilizează pachetul Microsoft 365, facilitând distribuirea materialelor didactice și comunicarea cu profesorii (Tan et al., 2022).
* Limitări: Aplicația este considerată de mulți utilizatori ca fiind resursivă, necesitând un consum semnificativ de RAM și CPU, în special pe dispozitive mai vechi (TechRadar, 2023). De asemenea, funcționalitatea de căutare a documentelor este limitată – nu permite întotdeauna căutarea exactă în conținutul fișierelor PDF sau Word încărcate pe canale.

**Evernote**

* Avantaje: Evernote este o aplicație de notițe avansată, care permite organizarea pe „notebook-uri”, adăugarea de etichete, sincronizarea pe mai multe dispozitive și funcția de OCR (recunoaștere text în imagini). Este utilă pentru centralizarea notițelor din cursuri, bibliografie și materiale proprii (Nast, 2021).
* Limitari: Planul gratuit este limitat la două dispozitive și 60 MB de upload pe lună. În plus, versiunile plătite sunt relativ costisitoare, comparativ cu alternative precum Notion sau OneNote. Un alt dezavantaj este **complexitatea interfeței**, care poate deveni copleșitoare pentru utilizatorii noi (Pot, 2023).

**Google Calendar**

* **Avantaje:** Google Calendar este o unealtă esențială în planificarea activităților academice. Se sincronizează cu Gmail, Google Meet și aplicații educaționale precum Google Classroom. Permite integrarea de evenimente recurente, memento-uri, obiective și partajare cu colegii sau profesor (Aldawsari et al., 2023).
* **Limitări:** Vizualizarea pe mobil poate deveni aglomerată în cazul unui program foarte încărcat. De asemenea, personalizarea culorilor sau categoriilor este limitată în aplicația nativă, necesitând extensii terțe. Pentru utilizatorii internaționali, **fusul orar poate genera confuzii** în cazul sincronizării cu sesiuni sau examene programate global.

**Aplicații Pomodoro / Aplicații de blocare a site-urilor (Forest, Focus To-Do, Freedom etc.)**

* **Avantaje:** Aceste aplicații sprijină **focusul și reducerea distragerilor** prin tehnici de gestionare a timpului precum metoda Pomodoro (25 min. lucru / 5 min. pauză) și prin blocarea accesului la aplicații sau site-uri precum rețelele sociale. Ele îmbunătățesc semnificativ productivitatea percepută (Kuhnle et al., 2012).
* **Limitări:** Pentru studenții care folosesc telefonul și pentru materiale de studiu (PDF-uri, aplicații educaționale), aceste aplicații pot deveni **restrictive sau impractice**. De asemenea, multe dintre ele pot fi ușor dezactivate sau ocolite, ceea ce reduce eficiența lor în cazurile de auto-disciplină scăzută (Rafiq et al., 2024).

**Google Drive**

* **Avantaje:** Google Drive este una dintre cele mai populare platforme de stocare în cloud, oferind 15 GB spațiu gratuit și integrare cu restul suitei Google (Docs, Slides, Sheets). Este folosit frecvent pentru a partaja materiale de curs, teme și proiecte în grupuri (Kim et al., 2021).
* **Limitări:** Spațiul gratuit este partajat cu Gmail și Google Photos, astfel încât mulți utilizatori ajung rapid la **limita de stocare**. Pentru extinderea spațiului, este necesar un abonament Google One. Există, de asemenea, **riscuri de confidențialitate** privind modul în care fișierele sunt scanate și indexate automat de către platformă (Electronic Frontier Foundation, 2025)

**Todoist**

* **Avantaje:** Todoist permite gestionarea eficientă a sarcinilor, folosind limbaj natural („scrie eseul vineri la 10:00”), etichete, proiecte și niveluri de prioritate. Are o interfață intuitivă și se sincronizează cu Google Calendar, Outlook și alte aplicații relevante pentru educație (Todoist, 2023)
* **Limitări:** Planul gratuit oferă doar funcționalități de bază. Pentru acces la filtre, memento-uri și funcții avansate, este necesar un abonament Premium. De asemenea, aplicația poate deveni **complexă în timp**, iar studenții pot pierde timp personalizând liste și etichete în exces.

## 1.3 Fragmentarea informației și impactul asupra productivității studentilor

În era digitală, studenții se confruntă cu o abundență de informație provenind din surse diverse: cursuri pe platforme universitare, e-mailuri, chat-uri de grup, fișiere partajate pe cloud și materiale de auto-studiu. În lipsa unui sistem coerent de integrare, această **fragmentare a informației** devine un obstacol major în procesul de învățare și organizare.

**Surse multiple, platforme multiple**

Majoritatea universităților folosesc astăzi platforme precum Moodle, Microsoft Teams, Google Classroom sau Canvas, alături de alte aplicații terțe pentru predare și colaborare. Deși aceste platforme facilitează accesul la resurse educaționale, ele creează și un mediu **fragmentat**, în care studenții trebuie să caute informații relevante în mai multe locuri (Diana Saplacan, 2019).

Într-un studiu efectuat la Universitatea din Bergen, studenții au raportat că informațiile academice sunt împrăștiate între emailuri, postări pe Teams, anunțuri pe LMS și mesaje de pe grupuri informale (WhatsApp, Discord). Această lipsă de centralizare creează confuzie, pierderi de timp și frustrare (Turel & Qahri-Saremi, 2016).

**Efecte asupra productivitatii**

Fragmentarea informației afectează în mod direct **productivitatea studenților**. Timpul petrecut căutând fișiere, deadline-uri sau instrucțiuni dispersate poate duce la scăderea motivației și a capacității de concentrare (Turel & Qahri-Saremi, 2016). Într-un studiu calitativ, 42 % dintre respondenți au declarat că pierd zilnic cel puțin 20 de minute încercând să regăsească resurse deja parcurse, ceea ce duce la o senzație generală de haos organizațional (Kim, 2021).

Pe termen lung, lipsa unui sistem coerent de agregare a informațiilor educaționale afectează capacitatea studenților de a construi o **imagine de ansamblu** asupra progresului lor, reducând eficiența planificării și învățării active (Fischer et al., 2020).

**Solutii posibile**

Pentru a combate fragmentarea, cercetările recomandă adoptarea unor sisteme de **centralizare a resurselor** (ex. folosirea unei singure platforme integrate de tip LMS) și instruirea studenților în metode de organizare digitală. Utilizarea unui planner digital legat de un sistem de notițe (ex. Notion, OneNote) permite gruparea cursurilor, fișierelor și termenelor într-un spațiu coerent și personalizat.

De asemenea, adoptarea unui „sistem de management al informației personale” (PIMS – Personal Information Management System) s-a dovedit eficientă în creșterea performanței academice, mai ales în contextul hibrid sau online (Jones, 2017).

## 1.4 Burnout academic – cauze, manifestări și consecințe

Burnout-ul academic reprezintă o stare de epuizare emoțională, mentală și fizică, asociată cu stresul cronic din mediul educațional. Acesta afectează tot mai mulți studenți, în special în contextul creșterii volumului de muncă, competiției academice și presiunii de a performa în paralel cu activități extracurriculare sau joburi part-time (Lin & Huang, 2014)

Principalele cauze identificate în literatura de specialitate includ suprasolicitarea academică, lipsa timpului pentru activități recreative, imposibilitatea de a controla programul de studiu sau metodele de predare, precum și presiunea constantă a performanței – fie autoimpusă, fie exercitată de mediul universitar sau familial. În plus, fragmentarea informației, multitasking-ul digital și ritmul accelerat al cursurilor contribuie la apariția unei stări de oboseală cognitivă persistentă. Studiile arată că izolarea socială, mai ales în contextul învățământului online, agravează aceste simptome, deoarece studenții se simt singuri în fața provocărilor academice (Salmela‐Aro & Upadyaya, 2014)

Manifestările burnout-ului sunt diverse, dar cele mai frecvente sunt epuizarea emoțională, pierderea motivației, dezinteresul față de activitățile academice, scăderea capacității de concentrare și deteriorarea performanțelor academice. Mulți studenți descriu senzații de gol interior, oboseală cronică și lipsă de sens, însoțite de iritabilitate, anxietate și uneori tulburări fiziologice, cum ar fi insomnia sau durerile de cap (Schaufeli et al., 2002). Aceste simptome pot fi ușor confundate cu depresia sau alte tulburări psihice, dar spre deosebire de acestea, burnout-ul are o legătură directă cu activitatea educațională și cu relația individului cu sistemul de învățământ.

Consecințele pe termen lung ale burnout-ului academic sunt semnificative. Printre acestea se numără abandonul școlar temporar sau definitiv, apariția unor afecțiuni psihosomatice, scăderea generală a calității vieții și diminuarea sentimentului de autoeficacitate – adică încrederea în propria capacitate de a progresa și a obține rezultate. În plus, mulți studenți care se confruntă cu burnout ajung să își piardă motivația intrinsecă pentru învățare, transformând procesul educațional într-o simplă obligație, lipsită de sens și satisfacție personală (Salmela-Aro & Read, 2017)

Prevenirea și gestionarea eficientă a burnout-ului implică intervenții atât la nivel individual, cât și instituțional. Este esențial ca studenții să învețe să își stabilească limite realiste, să își includă pauze regulate în programul zilnic și să adopte metode active de autoreglare emoțională și cognitivă. Jurnalele de reflecție, tehnicile de gestionare a timpului și strategiile de învățare autoreglată pot contribui la echilibrarea sarcinilor și menținerea unei stări de bine. În paralel, universitățile ar trebui să ofere servicii de consiliere psihologică accesibile și să creeze un mediu predictibil și flexibil, în care feedback-ul regulat, transparența evaluării și ritmul clar al predării să susțină sănătatea mintală a studenților (May et al., 2015)

## 1.5 Concluzii privind nevoia unei soluții integrate

Analiza modalităților actuale de organizare academică evidențiază un peisaj complex, în care studenții apelează la numeroase tehnici și instrumente digitale pentru a-și gestiona sarcinile, timpul, materialele și activitățile colaborative. Deși fiecare metodă analizată – de la planificarea calendaristică până la aplicațiile Pomodoro și utilizarea platformelor educaționale – aduce beneficii punctuale, se remarcă totodată o serie de **ineficiențe recurente**, precum fragmentarea ecosistemelor digitale, suprasolicitarea cognitivă, lipsa de integrare funcțională și dificultățile de menținere a motivației pe termen lung.

Utilizarea simultană a mai multor aplicații neconectate, absența unui calendar unificat care să integreze cursurile, examenele și proiectele, dificultatea de a accesa rapid resursele relevante sau lipsa unor mecanisme automate de urmărire a progresului contribuie la apariția unei stări de **dezorganizare informațională**. Acest fenomen, cumulat cu presiunea academică tot mai mare, favorizează apariția burnout-ului și afectează negativ performanțele academice și bunăstarea psihologică a studenților (Salmela‐Aro & Upadyaya, 2014)

Datele prezentate în subcapitolele anterioare susțin cu claritate nevoia unei soluții digitale integrate, care să unifice funcționalitățile de organizare academică într-un singur spațiu de lucru coerent. O astfel de platformă ar trebui să permită gestionarea eficientă a orarului personalizat, urmărirea temelor și proiectelor, centralizarea fișierelor didactice, sincronizarea automată cu notificări și, ideal, integrarea unor funcții de reflecție personală sau autoevaluare. În plus, ar fi esențial ca această soluție să fie intuitivă, compatibilă cu mai multe dispozitive și adaptabilă la nevoile individuale ale utilizatorilor.

În concluzie, fără o soluție centralizată și inteligentă de organizare academică, studenții vor continua să depindă de metode disparate, risipindu-și resursele mentale și emoționale în activități de coordonare în locul celor de învățare. O intervenție digitală bine gândită poate transforma această dinamică și poate deveni un factor decisiv în creșterea eficienței, motivației și sănătății mintale în mediul universitar contemporan.

# 2. Proiectarea de detaliu a aplicației informatice

## 2.1 Definirea obiectivelor aplicației informatice

Aplicația informatică propusă își propune să răspundă direct nevoilor identificate în capitolul anterior, prin oferirea unei soluții integrate de organizare academică, destinată studenților din învățământul superior. În contextul suprasolicitării digitale, fragmentării informaționale și creșterii incidenței burnout-ului academic, este esențială existența unui instrument unificat, eficient și accesibil, care să susțină atât planificarea activităților academice, cât și gestionarea resurselor și a timpului personal.

Scopul principal al aplicației este optimizarea modului în care studenții își gestionează sarcinile, evenimentele, materialele de curs și prioritățile zilnice, printr-o interfață intuitivă și adaptabilă. Soluția informatică va funcționa ca un „hub educațional personal”, în care utilizatorul va avea acces centralizat la toate elementele esențiale pentru parcursul academic.

Obiectivele concrete ale aplicației sunt:

* + **O1: Centralizarea informațiilor academice**, prin integrarea într-o singură platformă a orarului de cursuri, examenelor, temelor și proiectelor, cu posibilitatea de a adăuga și evenimente personale;
  + **O2: Automatizarea notificărilor și memento-urilor**, pentru termene limită, sesiuni de învățare sau întâlniri importante, astfel încât utilizatorul să fie permanent informat și să evite amânarea sarcinilor;
  + **O3: Organizarea materialelor de curs**, printr-un sistem de stocare și clasificare logică a fișierelor (PDF, prezentări, notițe), direct asociate cu evenimentele din calendar sau cursurile din orar;
  + **O4: Susținerea autoevaluării și reflecției**, prin funcționalități care permit monitorizarea progresului academic, marcarea sarcinilor finalizate și adăugarea de notițe personale pentru recapitulare sau planificare viitoare;
  + **O5: Asigurarea unei interfețe prietenoase și responsive**, adaptată utilizatorilor tineri, care poate fi utilizată pe dispozitive mobile, tablete sau desktop, cu accent pe claritate și simplitate;
  + **O6: Oferirea unui echilibru între activitățile academice și cele personale**, prin posibilitatea de a seta zone orare pentru odihnă, recreere sau activități extracurriculare – în spiritul prevenirii burnout-ului.

În ansamblu, aplicația urmărește să ofere un cadru de lucru coerent, personalizabil și ușor de utilizat, care să sprijine studenții în gestionarea eficientă a timpului și resurselor educaționale, contribuind astfel la creșterea performanței academice și la menținerea unei stări mentale echilibrate.

## 2.2 Proiectarea logică și fizică a ieșirilor

În cadrul unui sistem informatic orientat către utilizator, ieșirile reprezintă elementul final prin care aplicația comunică rezultatele procesării informației. În cazul aplicației propuse, ieșirile trebuie să fie clare, relevante, ușor de interpretat și să contribuie la luarea deciziilor organizatorice de către student. Acestea includ afișări în interfață, notificări automate, rapoarte vizuale și sinteze personalizate.

**Proiectarea logică a ieșirilor**

La nivel logic, ieșirile sunt construite pentru a răspunde direct cerințelor funcționale ale utilizatorului. Ele sunt grupate în mai multe categorii, în funcție de tipul de informație afișată:

* **Calendarul integrat** - afișează vizual cursurile, temele, examenele și evenimentele personale, folosind o structură lunară, săptămânală și zilnică. Fiecare eveniment conține titlu, descriere, interval orar, tipul activității și culoare asociată.
* **Lista de sarcini (To-Do List)** - generează automat sau manual sarcini pe baza cursurilor și proiectelor active, grupate pe zile sau săptămâni. Conține starea sarcinii (completată/neîncepută), termenul limită și prioritatea.
* **Notificări și memento-uri** - ieșiri temporale sub formă de alerte push sau e-mailuri care anunță apropierea unui deadline, începutul unui curs sau nevoia de revizuire a unui material.
* **Vizualizări de progres** - grafice și indicatori care arată numărul de sarcini finalizate, timpul total studiat într-o săptămână, gradul de respectare a planificării. Acestea susțin autoevaluarea și motivația.
* **Fișe de curs și materiale atașate** - sistemul permite afișarea fișierelor asociate fiecărui curs sau eveniment, cu opțiuni de descărcare, vizualizare și adăugare de notițe.

**Proiectarea fizică a ieșirilor**

La nivel fizic, ieșirile sunt realizate prin componente vizuale moderne, intuitive și responsive, astfel încât să poată fi accesate cu ușurință de pe orice tip de dispozitiv.

* Calendarul este implementat printr-un component de tip grid sau timeline interactiv, cu posibilitatea de filtrare a evenimentelor în funcție de categorie (examen, curs, deadline). Utilizatorul poate interacționa cu evenimentele pentru a vedea detalii sau a le modifica.
* Lista de sarcini este afișată într-o secțiune laterală sau dedicată, cu funcții de marcare ca finalizat, sortare după deadline și vizualizare a sarcinilor viitoare.
* Notificările sunt implementate folosind tehnologii de tip push notification (pe mobil) sau toast pop-up (pe desktop), cu posibilitatea de personalizare a frecvenței și tipului de alertă.
* Vizualizările de progres folosesc grafice circulare și bare generate cu biblioteci de vizualizare (ex. Chart.js, Recharts), fiind inserate în secțiunea „Profil utilizator” sau „Statistică săptămânală”.
* Fișierele și materialele didactice sunt gestionate într-un sistem de tip listă cu foldere, unde fiecare element poate fi deschis, descărcat sau comentat. Se utilizează componente de tip „accordion” pentru gruparea după cursuri.

Această abordare a proiectării ieșirilor are ca scop nu doar prezentarea informației, ci și crearea unui flux vizual logic și clar, care să susțină deciziile zilnice ale studentului fără a-l suprasolicita cognitiv. În plus, toate ieșirile respectă principiile designului accesibil și adaptiv, fiind optimizate pentru utilizatori cu diverse niveluri de experiență digitală.

## 2.3 Proiectarea logică și fizică a intrărilor

În cadrul unei aplicații informatice, proiectarea corectă a intrărilor este esențială pentru asigurarea unei interacțiuni eficiente și fluide între utilizator și sistem. Intrările reprezintă toate acele informații introduse de către utilizator în aplicație pentru a genera acțiuni, a crea conținut personalizat sau a actualiza datele deja existente. În cazul aplicației propuse, acestea includ evenimente în calendar, sarcini de lucru, note personale, fișiere încărcate, preferințe de notificare și date de autentificare.

**Proiectarea logică a intrărilor**

La nivel logic, intrările sunt structurate astfel încât să reflecte nevoile concrete ale utilizatorului și să fie coerente cu obiectivele aplicației. Tipologia principală a datelor de intrare este următoarea:

* Date privind evenimentele academice și personale, precum cursuri, examene, întâlniri sau pauze planificate. Acestea vor include titlul, descrierea, data și ora de început și sfârșit, categoria (ex. curs, examen, pauză), culoarea asociată și opțional, un fișier sau link.
* Sarcini și activități introduse manual sau generate automat din evenimentele academice, cu câmpuri pentru denumire, termen limită, prioritate, status și notițe.
* Date de autentificare și cont, care includ numele complet, adresa de e-mail, parola și opțional un număr de telefon sau o poză de profil.
* Materiale didactice și fișiere, care pot fi încărcate de utilizator pentru a fi asociate cu anumite cursuri sau evenimente.
* Notițe personale sau reflecții zilnice, introduse sub formă de text liber, care contribuie la componenta de autoevaluare și planificare.
* Preferințe de notificare, în care utilizatorul definește cum dorește să primească mementouri (push, e-mail), la ce oră și pentru ce tipuri de evenimente.

Toate aceste date vor fi validate înainte de procesare, pentru a evita conflictele logice, intrările incomplete sau redundante.

**Proiectarea fizică a intrărilor**

Proiectarea fizică a intrărilor presupune implementarea unor componente vizuale clare și intuitive care să faciliteze introducerea rapidă și precisă a datelor de către utilizator. Principiile de bază urmărite sunt minimalismul, claritatea semantică a câmpurilor și adaptabilitatea pentru mobil și desktop.

* Formularele de adăugare a evenimentelor folosesc componente de tip input text, dropdown, date picker și color picker, grupate logic în secțiuni clar delimitate (ex. detalii generale, notificări, atașamente).
* Introducerea sarcinilor se face printr-un formular rapid cu opțiuni de adăugare pe loc sau planificare ulterioară, cu selecție automată a termenului limită și categoriei în funcție de contextul calendaristic.
* Autentificarea și crearea contului sunt susținute de validare în timp real, folosind elemente de tip input controlat, protejat pentru câmpul de parolă, cu sugestii de securitate.
* Încărcarea fișierelor se realizează prin drag-and-drop sau selectare manuală, cu limită de dimensiune și extensii permise, iar fișierele sunt afișate imediat într-o listă cu opțiuni de redenumire și asociere cu un curs.
* Introducerea de notițe personale este simplificată printr-un editor de text rich, care permite formatări de bază (bold, italics, liste), astfel încât notițele să poată fi revizuite și reutilizate ulterior.
* Preferințele de notificare sunt gestionate prin toggle-uri și selectoare orare, permițând o personalizare ridicată fără suprasolicitarea interfeței.

Designul componentelor este realizat în concordanță cu standardele de accesibilitate (WCAG) și design responsiv, asigurând utilizabilitatea pentru studenți cu diverse nevoi tehnice și niveluri de experiență digitală.

Prin această abordare, sistemul încurajează introducerea corectă și completă a datelor, evitând erorile și simplificând experiența de utilizare, ceea ce contribuie decisiv la eficiența generală a aplicației.

## 2.4 Proiectarea sistemului de codificare a datelor

Sistemul de codificare a datelor are rolul de a asigura o structurare clară, logică și unitară a informațiilor în cadrul aplicației informatice. Acesta contribuie atât la eficiența prelucrării datelor în cadrul sistemului, cât și la identificarea rapidă și fără ambiguități a entităților gestionate (ex. utilizatori, evenimente, fișiere). Codificarea corectă este esențială pentru integritatea datelor, pentru optimizarea interogărilor în baza de date, dar și pentru menținerea consistenței în afișarea și utilizarea acestora în interfața aplicației.

**Principii generale ale codificării**

În cadrul aplicației, codificarea urmează următoarele principii:

* **Univocitate** – fiecare cod generat este unic în contextul entității sale (ex. fiecare utilizator are un ID unic, fiecare fișier atașat are o denumire distinctă);
* **Simplitate și lizibilitate** – codurile sunt concepute să fie cât mai ușor de interpretat pentru dezvoltatori și administratori, folosind convenții clare și delimitări logice;
* **Extensibilitate** – sistemul de codificare permite adăugarea de noi entități fără afectarea celor existente, susținând scalabilitatea aplicației;
* **Compatibilitate cu cerințele bazei de date relaționale** – codurile sunt alfanumerice sau numerice și pot fi folosite în relații între tabele ca chei primare sau străine.

**Exemple de codificare utilizate în aplicație**

* **Utilizator (User ID)**  
  Codul utilizatorului este generat automat și are forma U-YYYYNNN, unde YYYY reprezintă anul înmatriculării sau al înregistrării în aplicație, iar NNN este un identificator incremental (ex. U-2024035). Acesta asigură trasabilitatea utilizatorului și poate fi folosit în rapoarte statistice sau în monitorizarea progresului.
* **Eveniment academic (Event ID)**  
  Fiecare eveniment (curs, examen, întâlnire etc.) primește un cod de forma EV-TIP-XXX, unde TIP este abrevierea categoriei (ex. CRS pentru curs, EXM pentru examen, DLN pentru deadline), iar XXX este un număr incremental local pentru fiecare utilizator (ex. EV-CRS-001).
* **Fișier atașat (File ID)**  
  Fișierele încărcate sunt identificate printr-un cod unic generat automat, de forma F-USERID-SEQ.ext, unde USERID este codul utilizatorului, iar SEQ este numărul fișierului în ordinea încărcării (ex. F-U-2024035-02.pdf).
* **Sarcină (Task ID)**  
  Sarcinile introduse în to-do list sunt codificate după formatul T-YYYYMMDD-HHMMSS, marcând exact momentul introducerii pentru a evita conflictele (ex. T-20250511-134502).
* **Coduri de stare**  
  Pentru câmpurile ce descriu starea unei sarcini sau eveniment (finalizat, în curs, anulat etc.), se folosesc coduri standardizate (NEW, INP, CMP, DEL), utile atât în interfață, cât și în baza de date.

**Avantajele sistemului propus**

Sistemul de codificare propus aduce multiple beneficii. În primul rând, permite o identificare rapidă a fiecărei entități și susține integritatea referențială în baza de date relațională. În al doilea rând, codurile pot fi ușor înțelese de utilizatorii tehnici (dezvoltatori, administratori de sistem), dar și de utilizatorii avansați care doresc exporturi sau integrări. În plus, structura extensibilă face ca sistemul să poată susține creșterea numărului de utilizatori, evenimente și fișiere, fără a necesita revizuiri majore de arhitectură.

## 2.5 Proiectarea bazei de date

Baza de date reprezintă componenta fundamentală a aplicației informatice, responsabilă cu stocarea, actualizarea și regăsirea rapidă a datelor introduse și generate de utilizatori. O proiectare riguroasă a acesteia asigură atât integritatea și consistența informațiilor, cât și performanța sistemului în timpul rulării aplicației.

Aplicația propusă se bazează pe un model relațional, implementat în PostgreSQL, datorită fiabilității, flexibilității și compatibilității ridicate cu aplicațiile web moderne. În structura sa, baza de date reflectă entitățile logice identificate în faza de analiză și susține funcționalitățile prevăzute în cerințele funcționale.

**Modelul conceptual**

Modelul conceptual al bazei de date a fost definit pornind de la identificarea principalelor entități și a relațiilor dintre acestea. Entitățile esențiale sunt:

* **Utilizator (User)**: conține datele personale ale studentului (nume, email, parolă criptată, data înregistrării).
* **Eveniment (Event)**: reprezintă cursuri, examene, termene limită, activități personale.
* **Sarcină (Task)**: activități asociate cu evenimente sau create manual.
* **Fișier (File)**: materiale didactice încărcate și asociate cu evenimente sau cursuri.
* **Notificare (Notification)**: memento-uri automate generate pentru evenimente și sarcini.
* **Notiță personală (Note)**: reflecții sau informații introduse de utilizator, nepublice.
* **Setări cont (Settings)**: preferințele personale ale utilizatorului privind notificările și interfața.

**Modelul logic**

Pe baza modelului conceptual, s-au definit tabelele relaționale principale și câmpurile aferente fiecărei entități. Iată o parte relevantă din schema logică propusă:

* users(id, first\_name, last\_name, email, password\_hash, created\_at)
* events(id, user\_id, title, description, type, start\_datetime, end\_datetime, color, created\_at)
* tasks(id, user\_id, event\_id, title, deadline, priority, status)
* files(id, user\_id, event\_id, filename, file\_path, upload\_date)
* notifications(id, user\_id, event\_id, task\_id, message, notify\_at, sent)
* notes(id, user\_id, content, created\_at)
* settings(id, user\_id, notify\_email, notify\_push, theme\_preference)

Fiecare tabel include o **cheie primară** (id) și, acolo unde este cazul, **chei străine** (ex. user\_id, event\_id) pentru a asigura integritatea referențială. Coloanele de tip dată/timp (created\_at, deadline, notify\_at) sunt stocate în format timestamp cu fus orar.

**Normalizarea bazei de date**

Baza de date este proiectată conform regulilor de normalizare până la forma 3NF (forma normală a treia), ceea ce înseamnă:

* Fiecare tabel stochează un singur tip de informație;
* Nu există duplicare inutilă de date;
* Toate coloanele depind direct de cheia primară.

Această structură reduce redundanța și facilitează modificările ulterioare ale aplicației fără a compromite coerența datelor.

**Considerații suplimentare**

În implementarea bazei de date se vor aplica următoarele măsuri suplimentare:

* **Criptarea parolelor** utilizatorilor folosind algoritmi standard (ex. bcrypt);
* **Indexarea** câmpurilor frecvent accesate (ex. email, start\_datetime) pentru optimizarea interogărilor;
* **Constrângeri de integritate** (NOT NULL, UNIQUE, FOREIGN KEY) pentru a preveni inconsistențele;
* **Back-up regulat** al bazei de date pentru a preveni pierderile de date critice.

Prin această arhitectură, baza de date susține în mod eficient funcționarea aplicației și permite extinderea ei viitoare, fără a afecta performanța sau securitatea informațiilor gestionate.

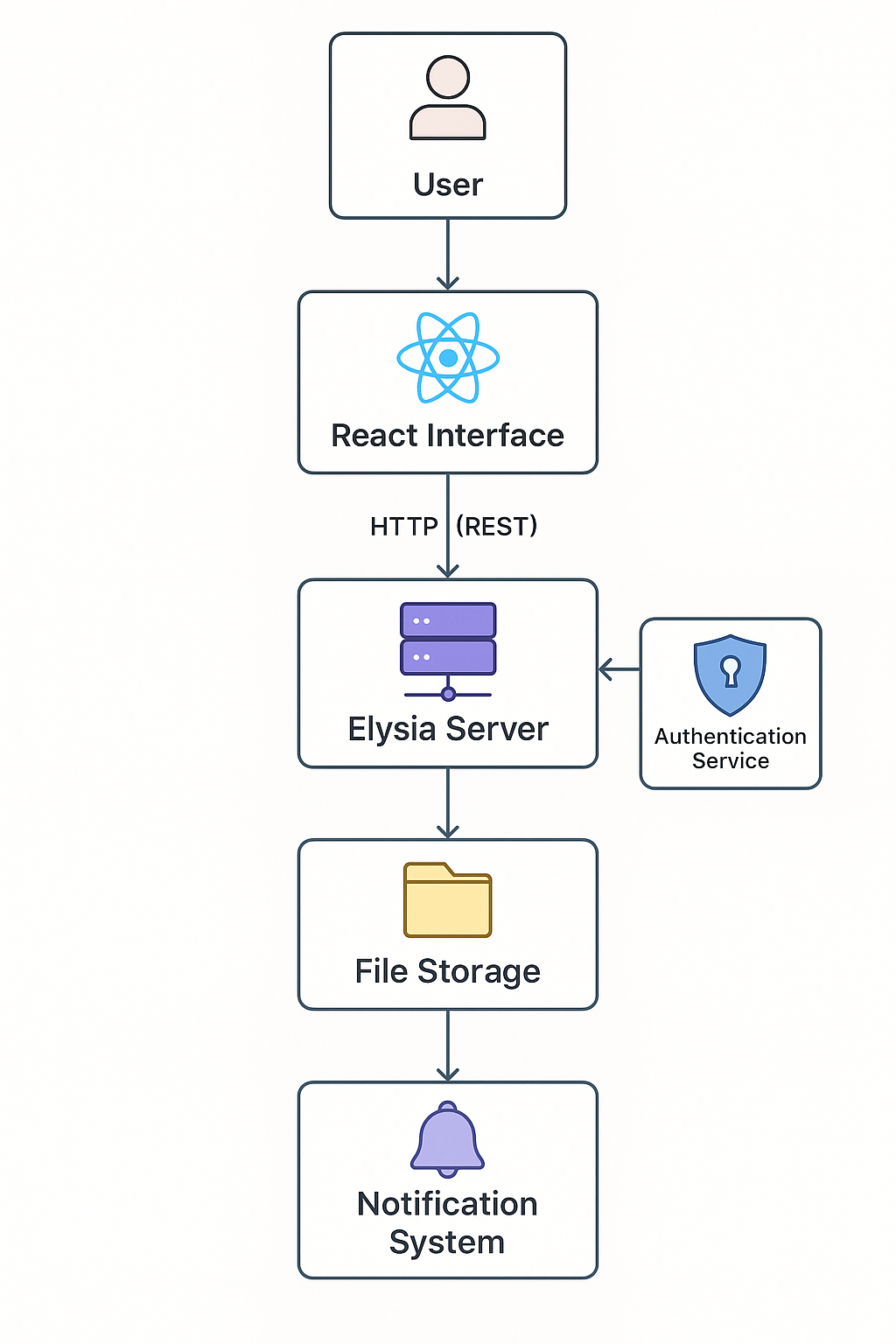
## 2.6 Schema de sistem a aplicației

Schema de sistem reprezintă o viziune de ansamblu asupra structurii aplicației, evidențiind componentele principale ale sistemului și modul în care acestea interacționează. Această reprezentare grafică și logică este esențială pentru a înțelege arhitectura internă a aplicației, precum și fluxurile de date între client, server și bazele de date.

Aplicația pentru organizarea academică a studenților este construită pe o arhitectură **client-server**, specifică aplicațiilor web moderne, cu următoarele componente majore:

**Componente ale sistemului:**

1. **Interfața utilizator (Frontend):**  
   Este partea vizibilă pentru utilizator, implementată folosind ReactJS și Tailwind CSS. Acesta permite introducerea și vizualizarea datelor (calendar, sarcini, fișiere, notificări) și oferă interacțiuni intuitive prin componente moderne (formular, drag-and-drop, to-do list, panou de control etc.).
2. **Serverul de aplicație (Backend):**  
   Reprezintă logica aplicației, implementată cu Elysia (sau alternativ ExpressJS). Acesta gestionează cererile HTTP primite de la client, validează datele, aplică logica de afaceri și comunică cu baza de date. De asemenea, serverul este responsabil cu autentificarea utilizatorilor și trimiterea notificărilor.
3. **Baza de date (PostgreSQL):**  
   Este responsabilă pentru stocarea tuturor informațiilor persistente ale aplicației: conturi de utilizatori, evenimente, sarcini, fișiere, notificări și preferințe. Comunicarea cu serverul se face prin ORM-ul Drizzle, care permite un acces sigur și optimizat la date.
4. **Serviciul de stocare fișiere:**  
   Fișierele atașate evenimentelor (ex. materiale de curs) pot fi stocate local sau într-un serviciu cloud precum Firebase Storage sau AWS S3. Serverul generează linkuri de acces pentru ca utilizatorul să poată vizualiza sau descărca fișierele.
5. **Sistem de notificări:**  
   O componentă care rulează în fundal (cron job sau webhook) generează notificări automate pe baza evenimentelor viitoare și a preferințelor utilizatorului. Acestea pot fi livrate sub formă de notificări push sau e-mail.
6. **Serviciu de autentificare:**  
   Integrează un mecanism de autentificare cu email și parolă, eventual extins cu login social (ex. Google), gestionând sesiuni și token-uri de securitate (JWT).



Această schemă ilustrează un flux clar: utilizatorul interacționează cu aplicația din browser, care comunică cu serverul pentru a salva sau extrage date, iar serverul le preia din baza de date sau din spațiul de stocare. Notificările sunt generate automat pe baza logicii din backend și livrate conform setărilor contului.

**Avantajele acestei structuri**

Structura modulară a sistemului permite:

* Scalabilitate: fiecare componentă poate fi extinsă sau înlocuită fără a afecta întregul sistem;
* Separarea clară între logică, prezentare și stocare;
* Întreținere facilă și posibilitatea de a adăuga noi funcționalități (ex. integrare cu calendare externe);
* Securitate mai bună prin izolarea componentelor sensibile (baza de date, fișiere, autentificare).

## 2.7 Proiectarea interfeței aplicației

Interfața aplicației (frontend-ul) reprezintă principalul punct de contact între utilizator și sistemul informatic. O interfață bine proiectată are rolul de a facilita introducerea datelor, înțelegerea funcționalităților și navigarea între module, contribuind în mod direct la experiența de utilizare și la eficiența activităților zilnice. În cazul aplicației informatice dedicate organizării academice, interfața trebuie să fie intuitivă, coerentă, estetic plăcută și adaptabilă la nevoile variate ale utilizatorilor – în special studenți din învățământul superior.

**Principii de design aplicate**

În procesul de proiectare a interfeței s-au respectat următoarele principii fundamentale:

* **Claritate vizuală** – folosirea de culori distincte pentru tipuri diferite de informație (ex. examene, cursuri, pauze), iconografie intuitivă și ierarhie tipografică bine definită;
* **Consistență** – toate paginile folosesc aceeași schemă de culori, stiluri de butoane și elemente interactive, creând o experiență unitară;
* **Accesibilitate** – contrastul text-fundal este adaptat pentru utilizatori cu deficiențe de vedere, iar toate funcțiile pot fi accesate atât prin mouse, cât și prin tastatură;
* **Responsivitate** – interfața este optimizată pentru afișare pe dispozitive desktop, laptop, tabletă și smartphone, fără pierderea funcționalității;
* **Reducerea sarcinii cognitive** – interacțiunea este ghidată prin elemente de feedback vizual, mesaje de eroare clare și completări automate (autocomplete, date picker).

**Structura principală a interfeței**

Aplicația este organizată în mai multe secțiuni, fiecare cu un rol clar în sprijinirea procesului de organizare academică:

* **Dashboard-ul principal** oferă o imagine de ansamblu asupra zilei în curs, incluzând: următoarele evenimente programate, sarcinile restante, notificările recente și un rezumat al progresului săptămânal. Este prima pagină accesibilă după autentificare.
* **Modulul de calendar** oferă o vedere lunară, săptămânală și zilnică, în care utilizatorul poate adăuga, edita și șterge evenimente. Se utilizează componente interactive de tip drag-and-drop pentru repoziționarea evenimentelor.
* **Modulul de sarcini (To-Do List)** permite gestionarea activităților zilnice, gruparea pe categorii (urgent, important, amânat) și marcarea progresului. Utilizatorul poate seta alerte pentru fiecare sarcină.
* **Secțiunea de materiale** permite încărcarea, clasificarea și vizualizarea fișierelor asociate fiecărui curs sau eveniment. Interfața suportă atașamente în diverse formate (.pdf, .docx, .pptx).
* **Profilul utilizatorului** conține setările contului, preferințele privind notificările, tema vizuală a aplicației (dark/light mode) și statisticile de activitate (ex. sarcini finalizate, ore de studiu înregistrate).

**Tehnologii utilizate**

Pentru dezvoltarea interfeței aplicației s-au folosit următoarele tehnologii:

* **React.js** – pentru structurarea componentelor UI, actualizarea dinamică a conținutului și gestionarea stării aplicației;
* **Tailwind CSS** – pentru stilizare rapidă și coerentă, folosind clase utilitare adaptabile;
* **Shadcn UI** – pentru componente predefinite (formulare, butoane, alerte) cu un design modern și personalizabil;
* **Framer Motion** – pentru tranziții animate care sporesc fluiditatea interacțiunii.

**Justificarea alegerilor de design**

Alegerea unui design minimalist, dar funcțional, are la bază profilul principal al utilizatorului: studentul care caută o aplicație rapidă, clară și adaptată ritmului său academic. Interfața propusă evită supraîncărcarea vizuală și oferă o experiență coerentă indiferent de platformă sau nivelul de competențe digitale.

## 2.8 Schema de flux informațional a noului sistem

Schema de flux informațional (data flow diagram – DFD) descrie modul în care informațiile circulă între componentele sistemului, utilizatori și procesele interne. Aceasta evidențiază relațiile logice dintre entități și oferă o viziune de ansamblu asupra mecanismului prin care datele sunt colectate, procesate, stocate și transmise în cadrul aplicației informatice.

În cazul aplicației pentru organizarea academică a studenților, fluxul informațional urmărește optimizarea interacțiunii dintre utilizator (student) și sistemul informatic, astfel încât activitățile zilnice să fie gestionate eficient, iar informațiile relevante să fie disponibile în timp real.

**Descrierea fluxului informațional**

Fluxul informațional poate fi descris în patru etape principale:

1. **Colectarea datelor**  
   Utilizatorul introduce manual date în aplicație prin intermediul interfeței grafice: evenimente academice, sarcini, notițe, fișiere sau preferințe de notificare. Totodată, unele date pot fi generate automat, cum ar fi notificările sau rapoartele de progres.
2. **Procesarea datelor**  
   Datele colectate sunt transmise serverului (backend), unde sunt validate, prelucrate și, dacă este cazul, transformate în noi obiecte: notificări programate, actualizări în calendar sau generarea unor rapoarte sintetice.
3. **Stocarea datelor**  
   Informațiile validate sunt salvate în baza de date relațională PostgreSQL (ex. conturi, evenimente, sarcini, notițe) sau în sistemul de stocare pentru fișiere (ex. documente de curs). Se respectă integritatea relațională și se evită redundanța.
4. **Furnizarea de ieșiri**  
   La fiecare acces al utilizatorului, sistemul generează vizualizări personalizate: calendarul zilnic, lista de sarcini, notificările programate și statistici privind progresul. Acestea sunt afișate în interfața React în timp real, prin apeluri API.

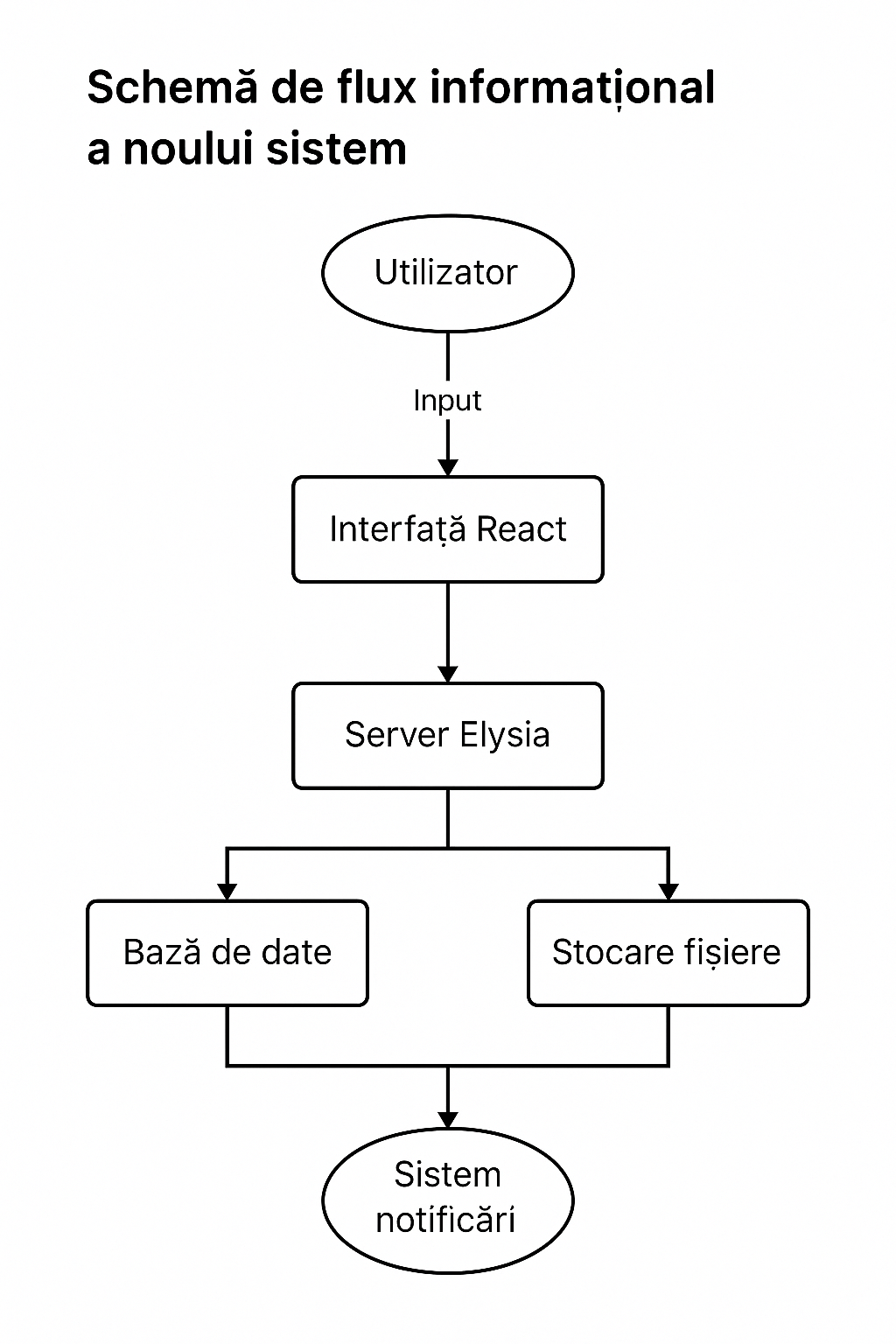
**Interacțiuni informaționale esențiale**

* Utilizatorul ↔ Interfață: introducere și afișare de informații;
* Interfață ↔ Server: schimb de date prin cereri HTTP (GET, POST, PUT, DELETE);
* Server ↔ Bază de date: salvare, modificare, interogare a datelor academice;
* Server ↔ Servicii externe: autentificare, stocare fișiere, trimitere notificări;
* Sistem notificări ↔ Utilizator: emiterea alertelor conform configurației individuale.

**Avantajele fluxului proiectat**

Fluxul informațional propus este **modular și eficient**, permițând:

* Coerență în circulația datelor între module;
* Minimizarea redundanței și a erorilor prin validare centralizată;
* Scalabilitate – fluxul poate fi extins pentru noi tipuri de date sau funcționalități;
* Timp redus de răspuns în interfață, datorită structurii bine definite a cererilor și răspunsurilor.



## 2.9 Alegerea tehnologiei de prelucrare

Alegerea tehnologiei de prelucrare este un pas esențial în proiectarea unei aplicații informatice eficiente, scalabile și ușor de întreținut. Această alegere trebuie să răspundă cerințelor funcționale și nefuncționale identificate anterior, să permită dezvoltarea rapidă a aplicației și să asigure o bună compatibilitate cu celelalte componente ale sistemului (interfață, bază de date, servicii externe).

Aplicația pentru organizarea academică a studenților este construită pe o arhitectură web modernă de tip **client–server**, în care prelucrarea datelor este realizată în principal pe partea de server (backend), dar cu o distribuție eficientă a sarcinilor și către client (frontend), acolo unde este necesar.

**Tehnologia aleasă: Elysia (Bun runtime) + JavaScript**

Pentru partea de server a aplicației, a fost aleasă tehnologia **Elysia** – un framework web performant, construit pe runtime-ul **Bun**, cu suport nativ pentru JavaScript. Această alegere este justificată prin următoarele avantaje:

* **Viteză de execuție foarte ridicată**, datorită motorului de rulare Bun, care este mai rapid decât Node.js în majoritatea operațiilor I/O și HTTP;
* **Sinteză modernă și concisă a codului**, care permite dezvoltarea rapidă și organizată a API-urilor REST;
* **Suport JavaScript nativ**, ceea ce aduce beneficii clare privind siguranța la rulare și completarea automată în timpul dezvoltării;
* **Integrare directă cu ORM-uri moderne** (ex. Drizzle) și biblioteci precum zod, cors, jwt, ceea ce permite validarea datelor, securizarea accesului și modularizarea logicii aplicației;
* **Performanță ridicată în producție**, cu latență redusă și consum minim de resurse, aspect esențial pentru aplicațiile web cu încărcare frecventă.

**Argumente suplimentare privind alegerea Elysia**

În comparație cu alte frameworkuri populare precum Express.js sau Fastify, Elysia oferă o combinație rar întâlnită între **viteză, tipizare și ergonomie**, fiind optim pentru proiecte noi, moderne, care au nevoie de o arhitectură clară și rapidă. De asemenea, comunitatea sa în creștere și documentația în curs de maturizare îl recomandă ca soluție viabilă pentru aplicații educaționale și MVP-uri scalabile.

**Structura prelucrării pe server**

Serverul Elysia va gestiona:

* Validarea datelor primite de la interfață (ex. formulare de evenimente, sarcini, preferințe);
* Prelucrarea logicii de afaceri (ex. generarea notificărilor, corelarea evenimentelor, prioritizarea sarcinilor);
* Trimiterea răspunsurilor către client în format JSON;
* Interacțiunea cu baza de date (PostgreSQL) și cu spațiul de stocare al fișierelor;
* Gestionarea autentificării și autorizării utilizatorilor (prin JWT).

Prin alegerea Elysia, aplicația beneficiază de un **cadru performant și extensibil**, capabil să susțină atât nevoile actuale ale proiectului, cât și eventuale extinderi viitoare (integrare cu alte platforme, funcționalități sociale, API public etc.).

## 2.10 Estimarea resurselor și calendarul de realizare

Realizarea unei aplicații informatice presupune o planificare atentă a resurselor necesare și stabilirea unui calendar realist de implementare. Această etapă permite identificarea cerințelor materiale, software și umane, precum și organizarea în timp a activităților, astfel încât dezvoltarea aplicației să se desfășoare coerent și predictibil.

**Estimarea resurselor necesare**

1. **Resurse umane:** Pentru dezvoltarea aplicației în forma sa actuală, este suficientă o echipă restrânsă, formată dintr-un dezvoltator full-stack, având cunoștințe solide de frontend (React, Tailwind CSS), backend (Elysia/Bun, JavaScript) și baze de date (PostgreSQL). În etape avansate, pot fi implicate și alte roluri (ex. designer UI/UX, tester QA).
2. **Resurse software:**

* IDE: Visual Studio Code (gratuit)
* Runtime: Bun + Elysia (open-source)
* SGBD: PostgreSQL (gratuit, open-source)
* ORM: Drizzle ORM
* Framework frontend: React
* Librării UI: Shadcn UI, Tailwind CSS
* Alte librării: Zod, JWT, Chart.js, Axios
* Servicii externe: Google Firebase (pentru stocare fișiere), Netlify/Vercel (pentru hosting frontend)

1. **Resurse hardware:**

* Laptop cu minim 8 GB RAM și procesor multicore;
* Conexiune stabilă la internet pentru testare și hosting;
* Spațiu de stocare minim 10 GB pentru dezvoltare locală și bază de date.

**Calendarul de realizare**

Planificarea proiectului se întinde pe o durată estimată de **8 săptămâni**, după cum urmează:

|  |  |  |
| --- | --- | --- |
| Săptămâna | Activitate principală | Detalii |
| 1 | Analiza cerințelor și documentarea | Identificarea funcționalităților, fluxurilor, tehnologiilor |
| 2 | Proiectarea bazei de date și a arhitecturii | Crearea modelului ER și a schemelor de sistem |
| 3 | Implementarea backend-ului (Elysia) – prima versiune | Rute pentru autentificare, evenimente și sarcini |
| 4 | Implementarea interfeței (React + Tailwind) | Calendar, dashboard, to-do list, fișiere |
| 5 | Conectarea frontend-backend + testare API | Testare locală, validări, gestionare erori |
| 6 | Implementarea notificărilor și autentificării securizate | JWT, memento-uri, setări cont |
| 7 | Testare completă + remedierea bugurilor | Testare funcțională și UX; îmbunătățiri UI |
| 8 | Publicare aplicație + redactare documentație | Deploy pe platformă de hosting + ghid utilizator |

**Observații finale**

Durata proiectului poate varia în funcție de complexitatea funcționalităților adăugate și de disponibilitatea resurselor. Totuși, calendarul propus este realist pentru un prototip funcțional (MVP), care poate fi ulterior extins. O bună organizare a timpului și utilizarea unor tehnologii moderne contribuie la optimizarea efortului și la reducerea întârzierilor.

# 3. Prezentarea produsului software

După finalizarea etapei de analiză și proiectare a aplicației, urmează prezentarea concretă a produsului software rezultat, din perspectiva funcționalităților implementate și a cerințelor tehnice necesare pentru rularea acestuia. Acest capitol își propune să ofere o imagine de ansamblu asupra aplicației dezvoltate, concentrându-se atât pe infrastructura necesară pentru instalare și utilizare, cât și pe descrierea detaliată a principalelor componente funcționale accesibile utilizatorului.

Scopul aplicației este de a oferi studenților o platformă digitală integrată care să le sprijine organizarea activităților academice, gestionarea eficientă a timpului și accesul rapid la resursele educaționale. Aplicația este concepută ca un instrument modern, intuitiv și adaptabil, capabil să funcționeze pe diverse dispozitive și să asigure o experiență de utilizare coerentă și sigură.

În cadrul acestui capitol vor fi detaliate, în subcapitolul 3.1, cerințele minime și recomandate ale platformei hardware și software necesare pentru utilizarea aplicației în condiții optime. Subcapitolul 3.2 va descrie funcționalitățile concrete ale aplicației, așa cum sunt ele percepute din perspectiva utilizatorului final, ilustrând modul în care fiecare componentă contribuie la atingerea obiectivelor definite în etapa de proiectare.

## 3.1 Cerințele platformei hardware și software ale produsului program

Pentru a asigura funcționarea optimă a aplicației informatice destinate organizării academice, este necesară definirea clară a cerințelor hardware și software aferente atât pe partea de client (frontend), cât și pe partea de server (backend). Aceste cerințe au fost stabilite în concordanță cu arhitectura aplicației și cu tehnologiile utilizate în procesul de dezvoltare.

**Cerințe pentru partea client (utilizator)**

Aplicația este accesibilă prin browser, nefiind necesară instalarea unui program dedicat. Pentru o experiență fluidă și completă, sunt recomandate următoarele:

**Cerințe hardware minime (dispozitiv utilizator):**

* Procesor: minim dual-core, 1.6 GHz
* Memorie RAM: minim 2 GB
* Spațiu de stocare: minimum 200 MB liberi pentru cache-ul browserului
* Rezoluție ecran: minim 1280x720 px

**Cerințe software minime (utilizator):**

* Sistem de operare: Windows 10+, macOS 10.13+, Android 10+, iOS 13+
* Browser compatibil: Google Chrome 100+, Mozilla Firefox 100+, Microsoft Edge 100+, Safari 13+
* Suport activ pentru JavaScript și CSS3
* Conexiune la internet stabilă

Aplicația este dezvoltată în React, ceea ce permite randarea eficientă a interfeței și compatibilitate largă între platforme. Interfața este complet responsive, adaptându-se atât la ecrane mari, cât și la dispozitive mobile.

**Cerințe pentru partea server (backend și hosting)**

Pentru rularea componentelor backend (scrise în Elysia/Bun + TypeScript), este necesar un mediu de execuție performant, stabil și compatibil cu serviciile externe de stocare și notificare.

**Cerințe hardware minime (server):**

* Procesor: 2 vCPU
* Memorie RAM: minim 2 GB
* Spațiu stocare SSD: minim 10 GB (pentru aplicație + baze de date)
* Lățime de bandă: minim 100 Mbps

**Cerințe software minime (server):**

* Sistem de operare: Linux (Ubuntu 20.04+ recomandat)
* Runtime: Bun (v1.0+)
* SGBD: PostgreSQL 14+
* Server HTTP: integrat sau configurat cu Nginx pentru reverse proxy
* Alte componente: Node/Bun package manager, suport TLS pentru conexiuni securizate, suport HTTPS

**Servicii externe integrate (opțional):**

* Firebase Storage / AWS S3 – pentru stocare fișiere
* Email API (ex. SendGrid) – pentru trimiterea notificărilor
* GitHub / GitLab – pentru versionarea codului și CI/CD

**Considerații privind portabilitatea și scalabilitatea**

Aplicația este concepută astfel încât să fie **portabilă** și ușor de migrat între diferite medii de rulare (de exemplu, de pe un server local pe un VPS sau în cloud). Arhitectura modulară permite extinderea ulterioară a funcționalităților sau migrarea către arhitecturi mai complexe (ex. microservicii), în funcție de nevoile viitoare.

## 3.2 Descrierea funcționalităților aplicației

Aplicația dezvoltată are ca scop central sprijinirea studenților în gestionarea coerentă a activităților academice printr-un instrument digital integrat, intuitiv și adaptabil. Aceasta oferă o experiență organizatorică completă, care îmbină planificarea timpului, urmărirea sarcinilor, stocarea materialelor de curs și autoevaluarea progresului academic, într-un spațiu unic și coerent.

În centrul aplicației se află procesul de autentificare, prin care fiecare utilizator își creează un cont personal protejat, folosind adresa de e-mail și o parolă criptată. După autentificare, utilizatorul este direcționat către un panou principal (dashboard), unde este întâmpinat de o sinteză clară a activităților zilei în curs, a sarcinilor restante și a notificărilor recente. Tot aici este afișat și un **countdown personalizat până la începerea sesiunii de examene**, calculat în funcție de datele introduse anterior în profil. Acest contor vizual îndeplinește atât un rol practic, cât și unul motivațional, ajutând utilizatorul să-și dozeze mai bine resursele de timp până la perioada evaluărilor.

Calendarul integrat al aplicației este unul dintre cele mai importante module funcționale. Acesta permite vizualizarea evenimentelor academice și personale într-un format lună, săptămână sau zi, oferind o perspectivă organizată asupra timpului. Spre deosebire de calendarele tradiționale rigide, aplicația propune un **orar complet customizabil**, în care utilizatorul poate introduce cursurile săptămânale, cu recurență automată, dar și activități neacademice precum pauzele, sesiunile de sport sau întâlnirile personale. Utilizatorul poate alege culori personalizate pentru fiecare tip de eveniment și poate grupa activitățile în funcție de priorități sau categorii tematice. Orice modificare este salvată în timp real, iar interfața de tip drag-and-drop permite repoziționarea rapidă a evenimentelor.

Sistemul de sarcini (to-do list) este perfect integrat în aplicație, astfel încât fiecare temă, proiect sau activitate planificată poate fi transformată într-o sarcină monitorizabilă. Utilizatorul poate vizualiza sarcinile ordonate cronologic, în funcție de termenul limită, și poate actualiza statusul acestora în funcție de progres. Sarcinile se pot genera manual sau automat, de exemplu atunci când se adaugă un eveniment de tip „deadline proiect”. Aplicația înregistrează progresul în mod automat și oferă feedback vizual privind gradul de realizare al obiectivelor, contribuind astfel la menținerea motivației.

O funcționalitate importantă este cea de stocare și gestionare a materialelor de curs. Fiecare eveniment din calendar poate avea atașate fișiere relevante, fie că este vorba despre suporturi de curs în format PDF, prezentări PowerPoint sau documente Word. Acestea sunt organizate în foldere virtuale, asociate cursurilor sau modulelor de studiu, și pot fi accesate, descărcate sau vizualizate direct din aplicație. Sistemul permite redenumirea fișierelor și adăugarea de notițe descriptive, facilitând căutarea ulterioară.

Un alt element important este sistemul de **notificări inteligente**, care funcționează atât sub forma de alerte în interfață, cât și prin e-mail, în funcție de preferințele utilizatorului. Notificările sunt generate automat înaintea evenimentelor marcate sau atunci când o sarcină se apropie de termenul limită. Această funcționalitate este complet configurabilă din secțiunea de profil, unde utilizatorul poate seta cât de devreme dorește să fie notificat, ce tipuri de evenimente să declanșeze alerte și în ce formă (vizual, push, e-mail).

Pentru studenții care își doresc o abordare autoreglată a învățării, aplicația pune la dispoziție o zonă de reflecție și notițe personale. Această secțiune permite introducerea de însemnări zilnice sau săptămânale, care pot conține idei, reflecții, autoevaluări sau liste de obiective. Utilizatorul poate accesa ulterior istoricul acestor notițe, ceea ce îi permite să observe evoluția propriului comportament academic și să-și ajusteze strategiile de învățare.

În paralel, aplicația oferă și o secțiune de analiză a progresului, unde sunt afișate grafice interactive și statistici relevante. Acestea includ numărul de sarcini finalizate versus restante, distribuția activităților în funcție de tip, timpul efectiv dedicat învățării sau participării la cursuri și frecvența utilizării aplicației. Datele sunt afișate sub formă de grafice de tip bară sau circular, astfel încât studentul să poată înțelege rapid unde se află și ce trebuie îmbunătățit.

Interfața aplicației este complet responsivă și a fost construită pentru a funcționa eficient pe desktop, tabletă și smartphone, fără a compromite accesibilitatea. Designul minimalist, dar funcțional, este însoțit de elemente de feedback vizual (animații, tranziții fluide, schimbarea culorii pentru evenimente active), care fac ca experiența de utilizare să fie atât eficientă, cât și plăcută.

Prin ansamblul funcționalităților sale, aplicația devine mai mult decât un simplu planificator: ea oferă un mediu complet de autoorganizare, reflecție, motivație și echilibru. Gândită ca un companion digital pentru întreaga perioadă universitară, aceasta răspunde nevoilor reale ale studenților și le oferă control asupra parcursului lor academic.

# 4. Eficiența și utilitatea aplicației informatice

Aplicația dezvoltată reprezintă o soluție digitală modernă, orientată spre îmbunătățirea modului în care studenții își gestionează activitățile academice și timpul personal. În acest capitol sunt analizate condițiile necesare implementării aplicației, modalitățile concrete de exploatare curentă și argumentele care susțin eficiența și valoarea practică a produsului informatic în contextul educațional actual.

## 4.1 Condiții privind implementarea aplicației

Implementarea aplicației se realizează în condiții tehnice accesibile și nu presupune o infrastructură hardware sau software avansată. Fiind o aplicație web, aceasta nu necesită instalare locală, iar toate componentele funcționează prin intermediul unui browser compatibil și a unei conexiuni la internet.

Pe partea client, aplicația este accesibilă de pe orice dispozitiv modern – laptop, desktop, tabletă sau smartphone – care îndeplinește cerințele minime prezentate în capitolul anterior. Utilizatorul nu are nevoie decât de un cont activ pentru autentificare și o conexiune stabilă pentru a beneficia de toate funcționalitățile.

Pe partea server, aplicația poate fi implementată fie pe o platformă de tip VPS, fie în cloud, folosind servicii gratuite sau comerciale precum Render, Railway, Vercel (pentru frontend) și PlanetScale sau Supabase pentru gestionarea bazei de date. Procesul de implementare presupune configurarea backend-ului (Elysia/Bun) și a bazei de date PostgreSQL, precum și publicarea frontend-ului React. Integrarea serviciilor externe, cum ar fi stocarea de fișiere sau trimiterea de notificări, se poate face gradual, în funcție de contextul de utilizare și bugetul disponibil.

Aplicația este scalabilă și modulară, ceea ce permite adaptarea ulterioară pentru mai multe categorii de utilizatori (ex. profesori, administratori), fără a fi necesară refactorizarea completă a arhitecturii.

## 4.2 Exploatarea curentă a aplicației

În utilizarea zilnică, aplicația acționează ca un asistent digital complet, în care studenții își pot planifica activitățile, pot stoca materiale, pot monitoriza termene și pot primi notificări legate de sarcinile și evenimentele importante.

Interfața prietenoasă și intuitivă permite utilizatorului să adauge rapid evenimente în calendar, să-și definească orarul și să configureze memento-uri automate. Funcția de countdown pentru sesiune adaugă un plus de conștientizare privind termenele academice și oferă o motivație vizuală constantă.

Modulul de sarcini se dovedește extrem de util în perioadele aglomerate, cum ar fi sesiunea de examene sau perioada predării proiectelor, permițând organizarea eficientă a timpului și prioritizarea activităților. Posibilitatea de a vizualiza progresul într-un mod grafic și de a reflecta asupra parcursului prin notițe personale aduce o componentă psihologică pozitivă, susținând autoreglarea și motivația intrinsecă.

Exploatarea curentă a aplicației presupune un minim de efort tehnic și timp, iar adaptarea la rutina zilnică a studentului se face natural, întrucât aplicația a fost gândită tocmai pentru a simplifica, nu a complica, organizarea academică.

## 4.3 Considerații privind eficiența aplicației informatice

Eficiența aplicației poate fi analizată din mai multe perspective: funcțională, tehnologică, psihologică și educațională.

Din punct de vedere funcțional, aplicația răspunde în mod direct unor probleme reale identificate în rândul studenților – lipsa unui spațiu centralizat de planificare, pierderea informației între platforme diferite, dificultatea în urmărirea progresului personal și riscul crescut de epuizare academică. Prin integrarea acestor funcționalități într-o singură aplicație coerentă, se obține o reducere semnificativă a timpului pierdut cu organizarea manuală și o creștere a controlului asupra propriei activități.

Tehnologic, aplicația este eficientă prin utilizarea unor soluții moderne, rapide și scalabile. Alegerea frameworkului Elysia cu runtime-ul Bun și a bibliotecii React pentru interfață asigură un timp de răspuns rapid, stabilitate și un consum redus de resurse. De asemenea, compatibilitatea cu servicii externe precum Firebase, Google Calendar sau sisteme de stocare în cloud permite extinderea facilă a aplicației.

Din perspectivă psihologică, aplicația contribuie la reducerea stresului asociat cu termenele limită, la creșterea motivației și la evitarea fenomenului de burnout. Prin mecanismele de reflecție și feedback vizual, studenții dezvoltă obiceiuri mai sănătoase de lucru și învață să își dozeze efortul.

În ceea ce privește impactul educațional, aplicația favorizează autonomia în învățare, planificarea eficientă și disciplina academică. Ea poate deveni un instrument valoros în completarea procesului educațional tradițional, în special în contextul actual, în care studenții trebuie să gestioneze simultan sarcini academice, activități extracurriculare și, uneori, locuri de muncă part-time.

În concluzie, aplicația informatică propusă nu doar că răspunde cerințelor inițiale ale proiectului, ci oferă o valoare reală și sustenabilă pentru utilizatorii săi, demonstrând eficiență atât în utilizarea resurselor, cât și în sprijinirea obiectivelor educaționale ale studenților.

# **Bibliografie**

Bolden-Barrett, V. (2019, January). *Workers with overstuffed to-do lists feel overwhelmed, not organized*. Retrieved from HRDIVE: https://www.hrdive.com/news/workers-with-overstuffed-to-do-lists-feel-overwhelmed-not-organized-study/546622/

Diana Saplacan, J. H. (2019). Use of Digital Learning Environments: A Study about Fragmented Information Awareness. *Interaction Design and Architecture(s) Journal*, pp. 86-109.

Electronic Frontier Foundation. (2025, April 27). *Student Privacy*. Retrieved from Electronic Frontier Foundation: https://www.eff.org/issues/student-privacy

Merkel, M. (2024, February 22). *Study Strategies: How to organize your time*. Retrieved from Randolph-Macon College wordmark: https://www.rmc.edu/news/how-to-organize-your-time/

Morgan, M. (2022, February 22). *Study Strategies: How to organize your time*. Retrieved from Randolph-Macon College: https://www.rmc.edu/news/how-to-organize-your-time/

Pot, J. (2023, November 29). *EverNote Review*. Retrieved from PC Mag: https://www.pcmag.com/reviews/evernote#:~:text=Evernote%20still%20works%20well%2C%20seemingly,to%20switch%20to%20an%20alternative.

Todoist. (2023). *Todoist Documentation*. Retrieved from Todoist: https://www.todoist.com/help

Aldawsari, H. K., Shahat, H. A., Gaber, S. A., Al-Ruwaili, H. A., Aldughaysh, A. M., & Mohmed, T. A. (2023). Developing Productive Thinking Skills in the field of Artistic Works using the SCAMPER Strategy for Twice Exceptional Students. *International Journal of Learning, Teaching and Educational Research*, *22*(12), 1–20. https://doi.org/10.26803/ijlter.22.12.1

Buehler, R., & Griffin, D. (2018). The planning fallacy. In *The psychology of thinking about the future.* (pp. 517–538). The Guilford Press.

Buehler, R., Griffin, D., & Peetz, J. (2010). The Planning Fallacy. In *Advances in Experimental Social Psychology* (Vol. 43, pp. 1–62). Elsevier. https://doi.org/10.1016/S0065-2601(10)43001-4

Kuhlthau, C. C. (2021). Guided Inquiry: Learning in the 21st Century. *IASL Annual Conference Proceedings*. https://doi.org/10.29173/iasl7647

Kuhnle, C., Hofer, M., & Kilian, B. (2012). Self‐control as predictor of school grades, life balance, and flow in adolescents. *British Journal of Educational Psychology*, *82*(4), 533–548. https://doi.org/10.1111/j.2044-8279.2011.02042.x

Lin, S.-H., & Huang, Y.-C. (2014). Life stress and academic burnout. *Active Learning in Higher Education*, *15*(1), 77–90. https://doi.org/10.1177/1469787413514651

May, R. W., Bauer, K. N., & Fincham, F. D. (2015). School burnout: Diminished academic and cognitive performance. *Learning and Individual Differences*, *42*, 126–131. https://doi.org/10.1016/j.lindif.2015.07.015

Rafiq, S., Iqbal, S., & Afzal, D. (2024). *The Impact of Digital Tools and Online Learning Platforms on Higher Education Learning Outcomes*.

Salmela-Aro, K., & Read, S. (2017). Study engagement and burnout profiles among Finnish higher education students. *Burnout Research*, *7*, 21–28. https://doi.org/10.1016/j.burn.2017.11.001

Salmela‐Aro, K., & Upadyaya, K. (2014). School burnout and engagement in the context of demands–resources model. *British Journal of Educational Psychology*, *84*(1), 137–151. https://doi.org/10.1111/bjep.12018

Salzer, M. S. (2012). A Comparative Study of Campus Experiences of College Students With Mental Illnesses Versus a General College Sample. *Journal of American College Health*, *60*(1), 1–7. https://doi.org/10.1080/07448481.2011.552537

Saplacan, D., Herstad, J., & Pajalic, Z. (2019). Use of Multiple Digital Learning Environments: A study About Fragmented Information Awareness. *Interaction Design and Architecture(s)*, *43*, 86–109. https://doi.org/10.55612/s-5002-043-005

Schaufeli, W. B., Martínez, I. M., Pinto, A. M., Salanova, M., & Bakker, A. B. (2002). Burnout and Engagement in University Students: A Cross-National Study. *Journal of Cross-Cultural Psychology*, *33*(5), 464–481. https://doi.org/10.1177/0022022102033005003

Steel, P. (2007). The nature of procrastination: A meta-analytic and theoretical review of quintessential self-regulatory failure. *Psychological Bulletin*, *133*(1), 65–94. https://doi.org/10.1037/0033-2909.133.1.65

Tan, C., Casanova, D., Huet, I., & Alhammad, M. (2022). Online Collaborative Learning Using Microsoft Teams in Higher Education Amid COVID-19: *International Journal of Mobile and Blended Learning*, *14*(1), 1–18. https://doi.org/10.4018/IJMBL.297976

Turel, O., & Qahri-Saremi, H. (2016). Problematic Use of Social Networking Sites: Antecedents and Consequence from a Dual-System Theory Perspective. *Journal of Management Information Systems*, *33*(4), 1087–1116. https://doi.org/10.1080/07421222.2016.1267529