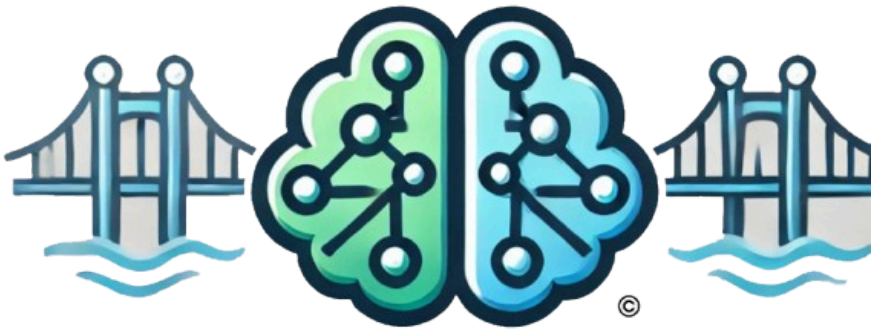
BRAUDE
College of Engineering, Karmiel

Software Engineering Department
Braude College

Capstone Project Phase B - 61999

Enhancing Classroom Engagement: A Multifaceted Tool for Managing Attention Difficulties



Knizhnik Sandra & Adar Nil

Mentor Dr.Natali Levi - Soskin

Repository: https://github.com/nil-adar/BrainB

Live System: https://brain-bridge.net/

Abstract

BrainBridge is a cross-platform system designed to empower students with attention-related challenges by delivering highly personalized support plans and fostering collaboration among learners, caregivers, and educators. This project is a direct continuation of the NODUS initiative, explicitly created to extend and integrate the diagnostic outputs produced by the NODUS system. At the heart of this project, which builds upon the NODUS diagnostic system, lies the integration of the NODUS model outputs, capturing each student's probability profile across ADHD(Attention Deficit Hyperactivity Disorder) subtypes. It also applies an eleven-stage recommendation pipeline to translate raw assessment data into concrete recommendations.

This pipeline consists of: Stages 1-6- ADHD subtype classification with parent input, Stage 7- Environmental context screening to distinguish neurodevelopmental from situational symptoms, Stages 8-11- Recommendation filtering and personalization.

The user interface guides each stakeholder through their respective questionnaire and then presents a dynamic recommendation dashboard. A dedicated tasks management module lets users schedule, track, and mark off daily interventions tailored to the student's profile. Meanwhile, a secure teacher-parent chat helps share updates and important information about the students.
Data persistence and business logic reside in an Express-based API connected to MongoDB, where diagnostic results, questionnaire responses, chat messages, and generated recommendations are all stored. This modular architecture ensures that BrainBridge can be deployed seamlessly across devices and scaled to support a wide range of diverse educational settings. In this report, we detail the system's architectural design, the implementation of its eleven-stage recommendation engine, and the development of its collaborative modules, highlighting how BrainBridge transforms initial diagnostic insights into actionable, stakeholder-driven intervention plans.

Key Words: ADHD, NODUS diagnostic model, Personalized intervention, eleven-stage filtering pipeline, Cross-platform application, Recommendation engine, Task management, Teacher-parent collaboration

Table of contents:

Introduction:

Attention related difficulties, especially those associated with ADHD, are increasingly prevalent in school age children and often impair focus, time management, and social interaction. Multiple factors contribute to these challenges, including neurological predispositions, environmental stressors, and the constant distractions generated by modern digital platforms such as TikTok and Instagram[1, 2]. Although several diagnostic and behavioral tools exist, such as the Behavior Rating Inventory of Executive Function (BRIEF)[3], the Child Behavior Checklist (CBCL)[4], and the Behavior Assessment System for Children (BASC)[5], these solutions were primarily designed for clinical settings. As a result, they require professional training and do not always translate into practical, day to day recommendations for classroom use[6].

NODUS is an external diagnostic platform that aggregates questionnaires, observations, and probabilistic models to produce subtype probability profiles for each student. While it meaningfully improves assessment precision, it does not translate its outputs into day to day, classroom ready interventions.

The second phase of this initiative, called BrainBridge, developed as an independent capstone project, builds on NODUS outputs by extending them into a holistic support system.

BrainBridge is a cross platform application that integrates diagnostic results with role specific workflows for teachers, parents, and students. At its core lies an eleven stage recommendation engine that classifies ADHD subtypes, incorporates contextual and environmental screening, applies evidence based filters, and generates personalized strategies across three key domains: environmental adjustments, nutritional interventions, and physical activity.

Beyond the recommendation engine, BrainBridge implements a task management module to help students structure daily routines, a secure teacher-parent chat for real time collaboration, and a PDF recommendation guide to provide a formal and shareable report. The system also underwent usability testing through the System Usability Scale (SUS), achieving an overall score of 83.75 ("Excellent"), which validated both the design and the user experience.

In this way, BrainBridge transforms diagnostic insights into practical and stakeholder driven action plans that evolve with the student and directly address the critical gap between clinical assessment and everyday classroom support[6, 7].

1. Literature Review

1.1 Introduction and Historical Background

ADHD is a neurodevelopmental condition characterized by persistent patterns of inattention, hyperactivity, and impulsivity that interfere with daily functioning. The concept of ADHD has evolved since the late 18th century, when Scottish physician Sir Alexander Crichton first described symptoms of inattentiveness and restlessness. Diagnostic criteria have shifted over time, from recognizing "Attention Deficit Disorder (ADD)" in 1980 to "ADHD" in 1987 [8], reflecting a deepening understanding of the disorder.

Today, ADHD affects 5–10% of school-aged children worldwide [9], with 60–70% continuing to experience symptoms into adulthood [10], corresponding to 4–7% of adults [11].

1.2 Neurobiological Insights and Environmental Influences

Research highlights differences in brain structure and function, particularly in regions linked to executive functioning, attention, and impulse control. These differences are reflected in dopamine and norepinephrine signaling [12]. At the same time, modern environmental pressures such as digital overstimulation have produced ADHD-like symptoms even in individuals without neurobiological ADHD. Frequent multitasking across apps and platforms fragments attention and disrupts reward pathways. Adolescents who spend more than six hours daily on digital media show significantly higher rates of attention difficulties[13] .

1.3 Impact of Lifestyle and Psychological Factors

Sleep disruptions, particularly due to blue light exposure from screens, impair melatonin production and double the risk of attention problems in adolescents [14]. Psychological stressors such as anxiety and depression also exacerbate attention difficulties, and nearly half of individuals with ADHD face comorbid conditions. This overlap emphasizes the need to address both mental health and attention regulation simultaneously [15, 16].

Attention challenges manifest differently across life stages. For children, difficulties often involve academic performance and peer rejection [17]. Adolescents face pressures from social media and heightened stress, which worsen concentration problems [18]. University students report deficits in time management, organization, and self-directed learning[19]. Adults experience challenges in workplace productivity, task management, and family life, requiring support strategies tailored beyond medication alone [20].

1.4 Gender Differences and Societal Impact

Gender significantly shapes ADHD presentation. Women are more likely to display inattentive symptoms, which can be masked, delaying diagnosis. They also have higher rates of co-occurring anxiety and depression [21]. Men tend to present disruptive hyperactivity and impulsivity, leading to earlier identification but also more frequent disciplinary actions [22]. Neurological studies show women often exhibit differences in emotional regulation areas, while men show more pronounced differences in motor control regions. Social norms further reinforce diagnostic disparities. Hormonal changes across puberty, pregnancy, and menopause exacerbate ADHD symptoms in women, highlighting the need for gender-sensitive diagnostic and treatment approaches [23, 24].

1.5 Treatment and Management Strategies

Pharmacological treatment remains foundational, targeting dopamine and norepinephrine systems and reducing symptoms for most patients [25, 26]. However, side effects, adherence issues, and access disparities complicate outcomes. Complementary strategies include nutritional support-minerals, omega-3, zinc, iron, magnesium, and vitamin D have shown potential benefits [27, 28, 29]. Physical activity, particularly aerobic exercise, enhances executive functions, working memory, and reduces hyperactivity, though adherence remains a challenge [30].

1.6 Gamification and Digital Tools

Gamification applies game elements such as challenges, rewards, and progress tracking to increase motivation and persistence. Studies show improvements in engagement and task completion among children with ADHD [31, 32]. In parallel, apps like Seesaw provide structured task organization and real-time teacher feedback, bridging the gap between diagnosis and classroom practice [33].

1.7 Design Thinking Approaches

Design thinking provides a user-centered methodology for developing interventions. By combining empathy, ideation, prototyping, and iterative testing, it enables the creation of innovative and practical solutions tailored to ADHD management [34].

1.8 Dopamine and Immune Links

Dopamine plays a central role in movement, motivation, and reward-related learning [35]. Beyond the nervous system, it influences immune regulation [36], suggesting that attention regulation may also intersect with broader physiological systems.

2 . General Description

Attention challenges - especially those associated with ADHD - are increasingly common among school-age children and can lead to difficulties with focus, time management, and organization in both academic and social settings. Despite the availability of diagnostic tools and individual learning aids, there remains a critical gap: no single platform unites students, caregivers, and educators to deliver coordinated, daily support and guidance.

The primary objective of the BrainBridge project is to design and implement a cross-platform system that seamlessly integrates diagnostic results with role-specific workflows, enabling personalized interventions and real-time collaboration. By combining automated recommendation generation with task scheduling and secure communication channels, BrainBridge seeks to improve student outcomes, reduce caregiver stress, and empower teachers with actionable insights.

The key user groups include:

- **Students** who receive daily task lists and visual cues to structure their study and behavior routines.
- **Parents/Caregivers**, who monitor progress, answer initial questionnaires, and stay informed through updates and chat messages.
- **Teachers/Educators**, who assign tasks, complete behavior reports, and engage in two-way communication to adapt interventions based on classroom observations.

3. Solution Description

3.1. Research Track Overview

In the research track, we conducted a thorough analysis of existing diagnostic and intervention tools for ADHD and attention challenges, focusing on four main areas: subtype classification algorithms, tag-based recommendation systems, multi-choice filtering methodologies, and evidence-based strategy selection. We reviewed peer-reviewed literature to identify validated approaches for mapping ADHD subtypes to targeted interventions and experimented with prototype filtering pipelines in Python notebooks. Key deliverables included: a comparative evaluation of classification accuracy for dominant subtype selection, a prototype tag-matching engine using sample questionnaire data, and documented trade-offs between exclusion-based versus prioritization-based recommendation ordering.

3.2. Development Track Overview

In the development track, we adopted an agile methodology with two-week sprints. Major implementation milestones included:

- Establishing the code repository and CI/CD pipeline.
- Implementing the Express.js API for CRUD operations on diagnostic results, answers, and recommendations.
- Developing React web and React Native mobile clients for role-specific questionnaires and the recommendation dashboard.
- Integrating the NODUS platform via RESTful hooks to import subtype probability data.
- Building the task management and chat modules with real-time updates using WebSocket for teacher-parent communication.
- Iterative usability testing sessions with sample users to refine UI flows and task visualizations.

3.3. Software Structure and Operation

BrainBridge is a web-based system designed to support communication and collaboration among teachers, students, and parents in the context of personalized education. The system streamlines the day-to-day tracking of tasks, observations, and diagnostic data, ensuring that each stakeholder has access to role-specific, relevant tools. For detailed technical architecture, see Section 4: Architecture Design.

3.3.1 System Workflow

At the core of BrainBridge's operation is a dynamic loop of interaction, observation, and adaptation that supports evidence-based personalization:

1. A **teacher** logs into the platform and sees their daily dashboard (screenshot below). They can select a class, add a student, schedule events, and assign a "Daily Task" form for each student.
2. If the task involves a diagnostic assessment, the teacher can activate the diagnostic process for a specific student. This triggers an automatic call to the NODUS platform, allowing the student (or their parent) to complete the assessment through their dedicated interface. Once completed, the system retrieves probability scores across ADHD subtypes and stores them in the database.

3. Recalculation & Caching Policy - The backend maintains a persistent cache of the **latest computed recommendations** per student `(StudentRecommendations)`.
   Before recomputing, the server checks whether there were **any changes** to student/parent/teacher forms or to the diagnostic result **since the last calculation timestamp**.

- If nothing changed → the API **returns the saved recommendations**, optionally filtered by the user's current `view` preference.
- If something changed (any form updated or a newer diagnostic exists) → the engine **recomputes** the full pipeline and upserts the result back to `StudentRecommendations`. This mechanism avoids unnecessary recomputation while keeping results fresh and consistent.

4. The **recommendation engine** runs in the backend and executes the following pipeline:
    - Detects the dominant subtype from the diagnostic result.
    - Matches questionnaire responses rated [3-4] to internal tags.
    - Applies exclusion filters (e.g., dietary constraints) and prioritization based on multiple-choice responses.
    - Assembles the most relevant recommendations in the environmental, nutritional, and physical activity categories.
5. These tailored recommendations appear in the teacher's dashboard, behind a 'View Recommendations' button. This functionality also exists for students and parents, each of whom sees the same set of recommendations relevant to the child. **Parents** can view related suggestions via their portal. A secure chat module enables two-way communication between teachers and parents to share observations and coordinate support.
6. **Students** view their assigned tasks via a simplified interface. Tasks may include focused learning activities, self-regulation prompts.

All data flows are persisted in MongoDB. Real-time updates (e.g., messages or task completions) are powered by WebSocket, while filtering operations are handled asynchronously using Node.js promises to ensure responsiveness. Behind the scenes, each user action (e.g., submitting a form or completing an assessment) triggers backend API endpoints that store responses, fetch diagnostic results, and return personalized recommendations for display. Below is a simplified operational flow diagram illustrating how key system components interact:

Together, these modules form a cohesive cycle that transforms assessment data into everyday action. BrainBridge empowers educators and families with personalized, practical tools to support students with attention-related challenges.

Refer to section 3.4 for a detailed explanation of the recommendation engine and filtering algorithm.

3.4. Algorithms and Methodology

The recommendation engine implements an eleven-stage pipeline that classifies ADHD subtype(s), screens for environmental factors, filters by role-specific tags, removes allergy conflicts, and then respects the user's display preference (main vs. both types).

Main vs. Both Types (Display Modes)

Each student receives a diagnostic classification: Hyperactivity, Inattention, Impulsivity, Combined, or No ADHD.

- **Single type** - if only one diagnosis type is identified, recommendations are shown for that type only.
- **Multiple types** - if more than one type passes the literature-based thresholds, the system offers a choice of display mode:
    - **Main view** → show recommendations only for the dominant type (the "main" diagnosis).
    - **Both views** → show recommendations for all identified types together.

If the student has only one type, the system does **not** ask and simply shows the main recommendations.

Thresholds used in code

- `MIN_NO_ADHD_VAL = 0.70` → ensures high confidence before classifying "No ADHD"
- `MIN_INATT_VAL = 0.20` → inattentive symptoms must reach a minimal clinical significance
- `MIN_HYPER_VAL = 0.25` → hyperactivity/impulsivity must reach a minimal clinical significance

These thresholds reflect clinical decision-making principles that balance diagnostic sensitivity with specificity. For detailed justification, see Section 11: Research-Based Threshold Justification.

Visual breakdown of stages

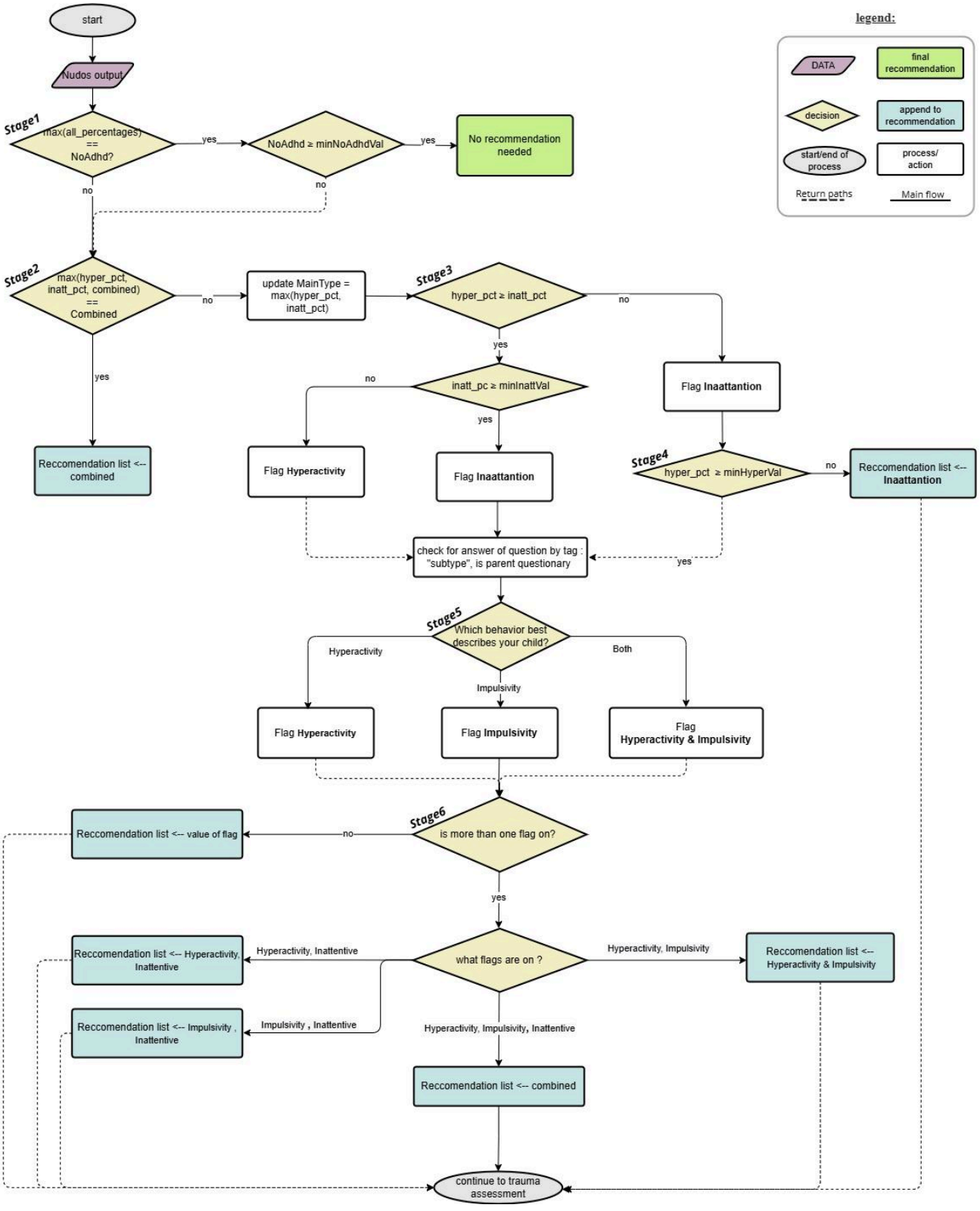
Stages 1-6: Diagnosis classification

At this phase, the system determines which subtype(s) are clinically relevant. The order reflects a "funnel logic": decisive exits are applied first (Stage 1), then dominant categories are established (Stages 2-3), parent input is incorporated (Stages 4-5), and results are consolidated (Stage 6).

Inputs from NODUS app:

- `combined_pct` - % of Combined detected
- `hyper_pct` - % of Hyperactivity detected
- `inattention_pct` - % of Inattention detected
- `noAdhd_pct` - % of No ADHD detected

Figure 1. Primary ADHD Classification Pipeline



Stage 1: Initial Screening Filter

- Action: If `max(all_pct) = noAdhd_pct` and `noAdhd_pct ≥ 0.70`, return "No recommendations needed" and terminate.
- Rationale: Ensures clear "No ADHD" cases exit early, avoiding unnecessary processing.

Stage 2 - Combined vs. Subtype Differentiation

- Action: If `combined_pct` is the maximum among {combined, hyper, inatt}, classify as Combined. Otherwise, continue with the dominant of {hyper, inatt}.
- Rationale: Combined is decisive-if it dominates, there is no need to compare further.

Stage 3 - Subtype Pathway Selection

- ***Actio***n: Set `dominantFromPercentages = (hyper_pct ≥ inatt_pct ? "Hyperactivity" : "Inattention").`
  - If Hyperactivity dominates and `inattention_pct ≥ 0.20` → also flag `Inattention.`
  - If Hyperactivity dominates and `inattention_pct < 0.20` → flag only `Hyperactivity.`
  - If `Inattention` dominates → flag `Inattention`.
- Rationale: Captures cases where Hyperactivity is stronger but Inattention still clinically relevant.

Stage 4 - Secondary Hyperactivity Check & Parent Trigger

- Action:
  - On the Inattention path: check if `hyper_pct ≥ 0.25`. If yes → trigger parent question q2-16.
  - On the Hyperactivity path: always trigger q2-16.
- Rationale: Parent input is considered only if Hyperactivity is clinically meaningful (≥25%), reducing noise from weak signals.

Stage 5 - Parent Disaggregation Protocol (q2-16)

- Action: Parent selects description:
  - opt1 → add Hyperactivity
  - opt2 → add Impulsivity
  - opt3 → add Hyperactivity & Impulsivity
  - opt4 → no additional flags
- Rationale: Parent insights refine classification, capturing behaviors not always obvious in algorithmic scoring (e.g., impulsivity).

Stage 6 - Multi-Flag Consolidation Logic

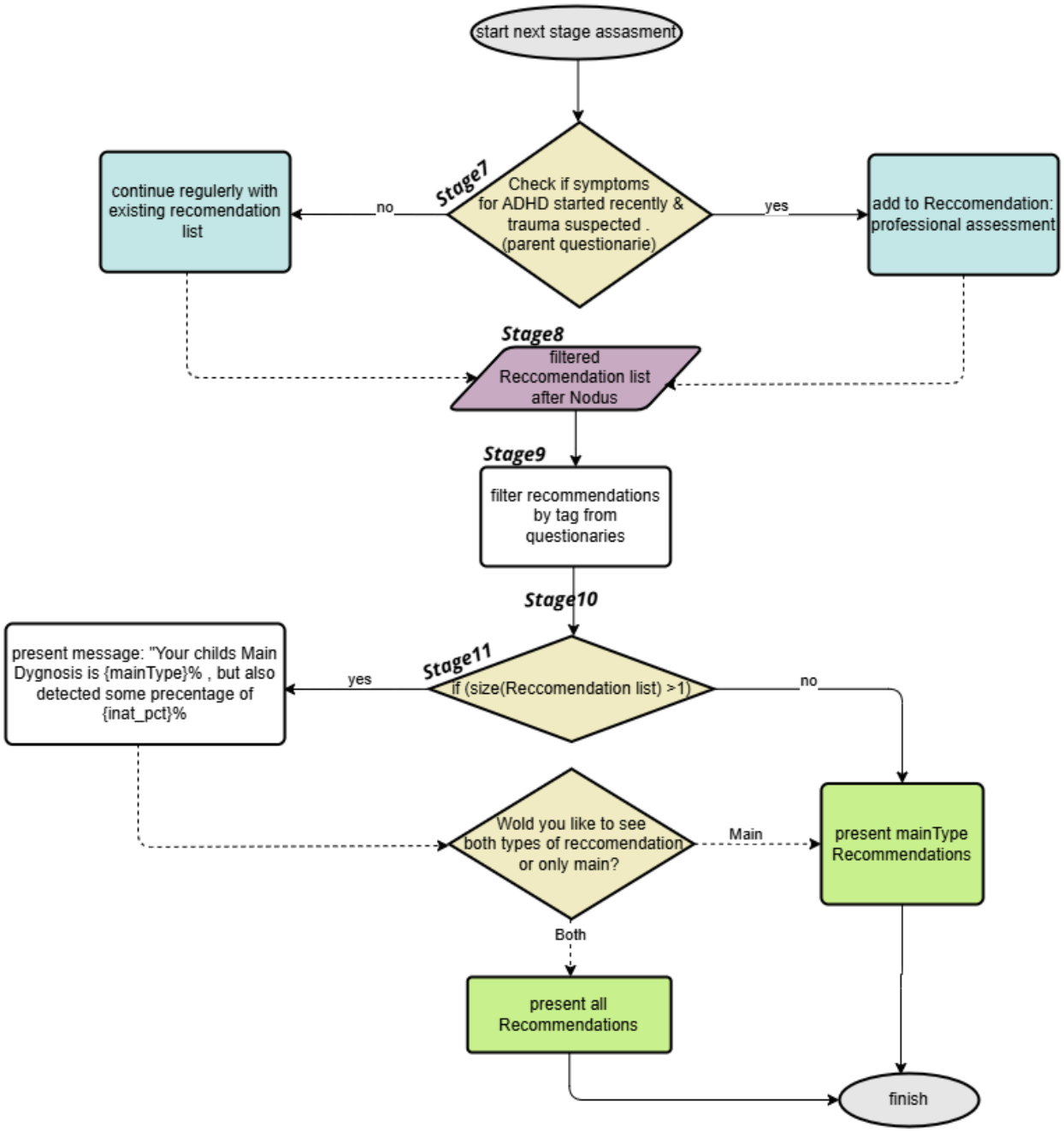
- Action: Combine all active flags.
    - One flag → single subtype.
    - Multiple flags → use dominance rule, unless the parent explicitly chose Impulsivity (which overrides as mainType).
    - All three flags → force Combined.
- Rationale: Ensures consistent, mutually exclusive classification while respecting parent overrides**.**

Design Note - Hyperactivity vs. Impulsivity

While the NODUS diagnostic model outputs hyperactivity and impulsivity as a single combined construct, BrainBridge separates them. This disaggregation step, implemented in Stages 4-6, was introduced to ensure that impulsivity is not overlooked when hyperactivity dominates. The rationale and supporting literature are detailed in Section 2.4.3.

Proceed to Figure 2 (Environmental Context Assessment Pipeline)

Figure 2. Environmental Context Assessment Pipeline



Stages 7-11: Environment, Tags, Allergy, and Display

At this phase, the system has established the primary ADHD classification from Stages 1-6. Now it processes environmental context, applies recommendation filtering, and prepares the final output for users.

Stage 7 - Environmental Onset Screening

- *Action*: Use q2-27 ("Have behaviors always been present?") and q2-28 ("Special event before symptoms?") to check whether behaviors are lifelong or triggered by recent environmental changes. If recent onset with a specific trigger, set `professionalSupport = true.`(this will trigger later on a professional support recommendation)
- *Rationale*: Distinguishes between neurodevelopmental ADHD and situational ADHD-like symptoms requiring therapeutic intervention.

Stage 8 - Filter by Diagnosis Type

- *Action*: Keep only recommendations whose `diagnosis_type` matches the student's main/subtypes.
- *Rationale*: Ensures recommendations align strictly with diagnostic classification outcome.

Stage 9 - Tag Filters (Difficulties & Special Cases)

- Action: Collect tags from questionnaire responses (answers 3 or 4). Always include trauma/professional tags, even if filtered elsewhere.
- Rationale: Captures significant difficulties while guaranteeing clinical safety recommendations surface regardless of other filters

Stage 10 - Allergy Filtering

- Action: For each allergy tag, remove only conflicting items from recommendation examples. Map allergy codes to food items using allergyMapping. For each recommendation's example arrays, remove items matching allergyList.
- Rationale: Maintains recommendation validity while preventing potentially harmful dietary advice.

Stage 11 - Presentation & User Preference

- Action:
    - If recommendationTypesList.length > 1: set multipleTypes=true, mainType=first element, subTypes=remaining elements
    - Apply view parameter: if view="main" → filter to mainType + special recommendations only
    - If view="both" or undefined → include all recommendations
    - Cache results in StudentRecommendationsModel with timestamp
- Rationale: Provides user flexibility between focused (main type) vs comprehensive (all types) views while ensuring special clinical recommendations always appear.

Output: Final personalized recommendation set ready for display, with proper caching for subsequent requests until forms/diagnostics are updated.

3.4.1 Core Philosophy

BrainBridge does not invalidate ADHD recommendations when environmental flags are raised. Instead, it supplements them with professional guidance (e.g., referral to therapist, blood tests). This ensures neurodevelopmental and situational factors are both addressed.

If `environmental_factors = true`, the system:

- **Maintains all existing ADHD recommendations** derived from NODUS analysis
- **Supplements** them with professional assessment guidance:
    - *Consult a psychological therapist* for a comprehensive evaluation
    - *Conduct blood tests* to rule out physiological contributors
- **Presents a dual-approach recommendation set** that addresses both ADHD-related and environmental concerns

3.4.2 Enhanced Design Rationale

This enhanced pipeline integrates **environmental context awareness** with **evidence-based ADHD assessment protocols**, co-developed with clinical practitioners specializing in pediatric attention disorders.

3.4.3 Clinical Insight Integration

Professional consultations revealed that numerical models alone cannot fully capture the complexity of ADHD presentations. Sudden changes in environment - such as trauma, relocation, or family disruptions - can lead to ADHD-like symptoms that require therapeutic intervention, even if not rooted in neurodevelopmental origins.

Motivation for Stage 7 (Environmental Onset Screening).
During an expert interview with **Liora Gaz**[37], we sought insights to refine the recommendation process. As Gaz does not work with medication but focuses on non-conventional, context-driven interventions, her perspective is aligned with the core ideology of BrainBridge.

She emphasized that attention difficulties are rarely “black and white” and are strongly influenced by life circumstances. According to her, it is crucial to bridge numerical diagnosis with the student’s emotional and situational context, since recent life events can trigger ADHD-like behaviors that should not be misinterpreted as neurodevelopmental ADHD.

Based on this insight, we introduced **Stage 7 - Environmental Onset Screening**. This stage explicitly checks whether symptoms appeared only recently and whether they are linked to a specific event. If so, the system appends a **professional-support recommendation** (e.g., referral to an emotional/therapeutic intervention) in addition to the standard ADHD-based recommendations.

This adjustment ensures that BrainBridge does not overlook situational factors and better reflects the complexity of real-world classroom dynamics, thereby reducing the risk of misattribution and supporting more effective, holistic interventions.

3.4.4 Hyperactivity-Impulsivity Disaggregation

The NODUS diagnostic model treats hyperactivity and impulsivity as a single merged construct, reflecting common research practices that operationalize these dimensions jointly. However, during Phase A of our project, literature review and academic consultation revealed that hyperactivity and impulsivity demonstrate meaningful differences in emotional correlates, treatment implications, and behavioral manifestations [38][39][40].

Research indicates that hyperactive/impulsive symptom dimensions relate specifically to emotional impulsivity patterns, while inattentive dimensions show distinct associations with emotion dysregulation challenges [38]. Furthermore, clinical literature emphasizes that hyperactive and impulsive presentations require different treatment approaches, supporting their conceptual separation for intervention planning [39].

To increase recommendation precision, BrainBridge disaggregates these constructs through parent-based questionnaire input (q2-16) within Stages 4-6 of our pipeline. This allows explicit classification into hyperactivity, impulsivity, or both, ensuring that impulsivity-specific recommendations are not overshadowed by hyperactivity scores.

This methodological extension beyond baseline NODUS logic reflects evidence-based understanding of ADHD's multidimensional nature [40], enabling more tailored and subtype-specific recommendations that align with contemporary ADHD research.

3.4.5 Research-Based Threshold Justification (summary)

Our threshold selection prioritizes diagnostic sensitivity over specificity, reflecting clinical evidence that ADHD underdiagnosis poses greater risks than overdiagnosis. Complete rationale and evidence base are detailed in Section 11.

3.4.6 Implementation Architecture

Conditional Logic Processing

- JavaScript conditionally renders q2-28 only if q2-27 indicates recent onset
- Environmental flags are processed in tandem with the tag-matching algorithm
- Professional assessment recommendations are appended to, not overriding, core NODUS-derived results

Data Management

- Environmental flags, tags, and exclusions are stored in MongoDB
  Logic runs asynchronously using Node.js promise chains
- All environmental indicators are logged for clinical traceability

Quality Assurance

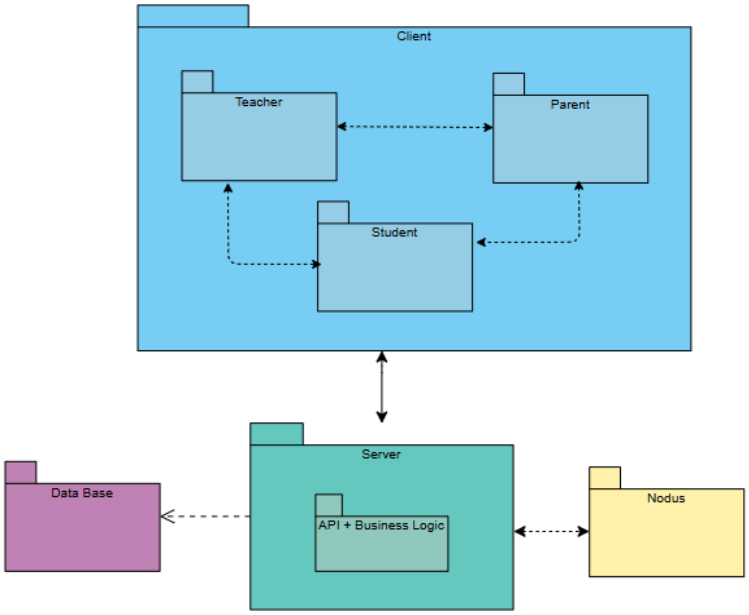
- Unit tests cover edge cases: tied subtype probabilities, empty tag sets, environmental flag combinations
- Integration tests confirm correct questionnaire flow and supplemental recommendation logic
- Performance tests ensure environmental screening has no adverse impact on responsiveness

This enhanced architecture preserves the evidence-based rigor of the original NODUS model while incorporating critical real-world context. It empowers the system to generate more nuanced, clinically informed recommendations, supporting both accurate diagnosis and holistic intervention planning.

4. Architecture Design

4.1 High-Level Architecture (Logical & Technical Views)

Figure 3: System Architecture Overview (Client-Role Interaction + Server Logic)



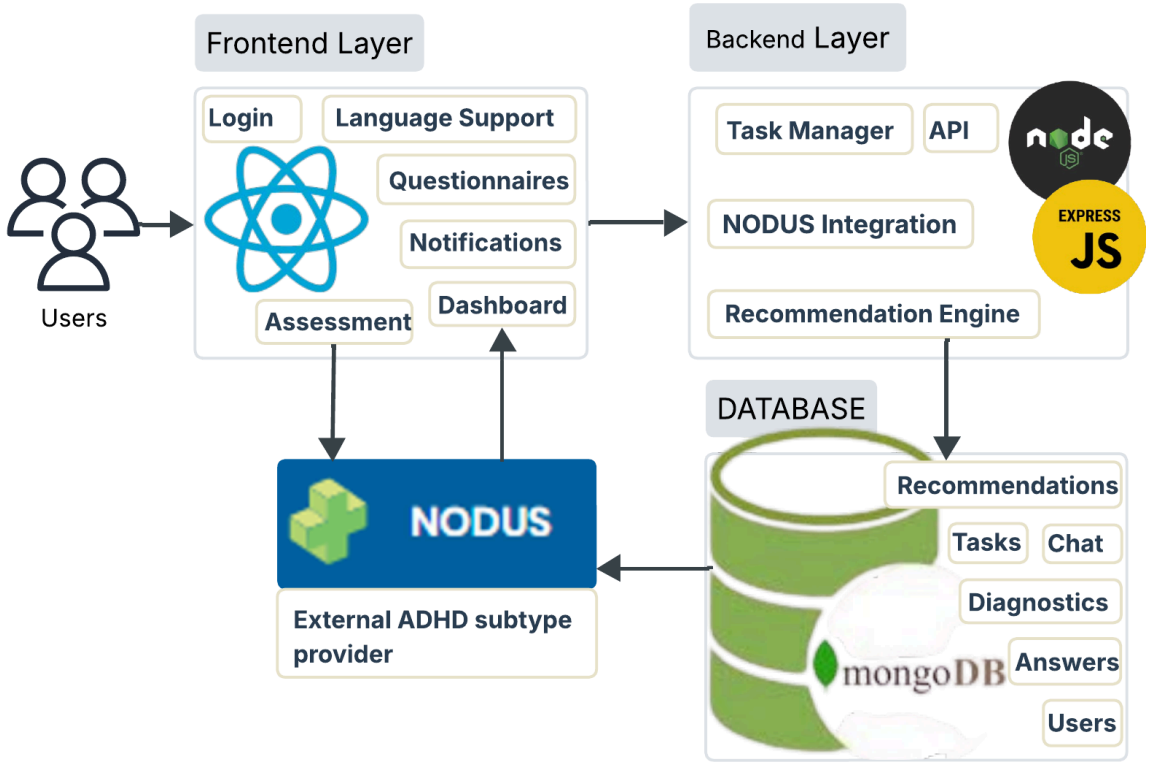
This diagram illustrates the logical flow and interaction between the system's core user roles and backend services. The Client package includes three distinct interfaces: Teacher, Parent, and Student, each with tailored functionality:

Teachers create and assign tasks, fill in observation forms, and trigger diagnostic assessments. Parents complete questionnaires, view recommendations, and engage in two-way communication with teachers. Students access daily tasks and participate in the NODUS diagnostic through a simplified UI. All communication passes through the central Server, which handles authentication, routing, and data processing. When a diagnostic is initiated, the server communicates directly with NODUS, retrieves results, and stores the output in the Database.

This architecture ensures clear separation of concerns and encapsulated role-based logic while supporting real-time feedback and synchronized collaboration.

While Figure 3 demonstrates the user-centric workflow and role-based interactions, Figure 4 details the underlying technical infrastructure and components that enable these interactions.

Figure 4: Technical Architecture & System Components



Frontend Layer:

- React-based interface providing role-specific functionality including Login, Language Support (Hebrew/English with RTL/LTR), Questionnaires, Notifications, Dashboard, and Assessment components

- Users (Teachers, Parents, Students) interact with tailored interfaces through the centralized React application
- Assessment components communicate with NODUS for diagnostic completion and results retrieval

Backend Layer:

- Node.js with Express.js serving as the core server technology stack
- Task Manager handles assignment creation, distribution, and completion tracking with timer functionality
- REST API manages all client-server communications and data exchange
- NODUS Integration service facilitates external diagnostic requests and response processing
- Recommendation Engine generates personalized suggestions based on assessment results and user data

Data Persistence:

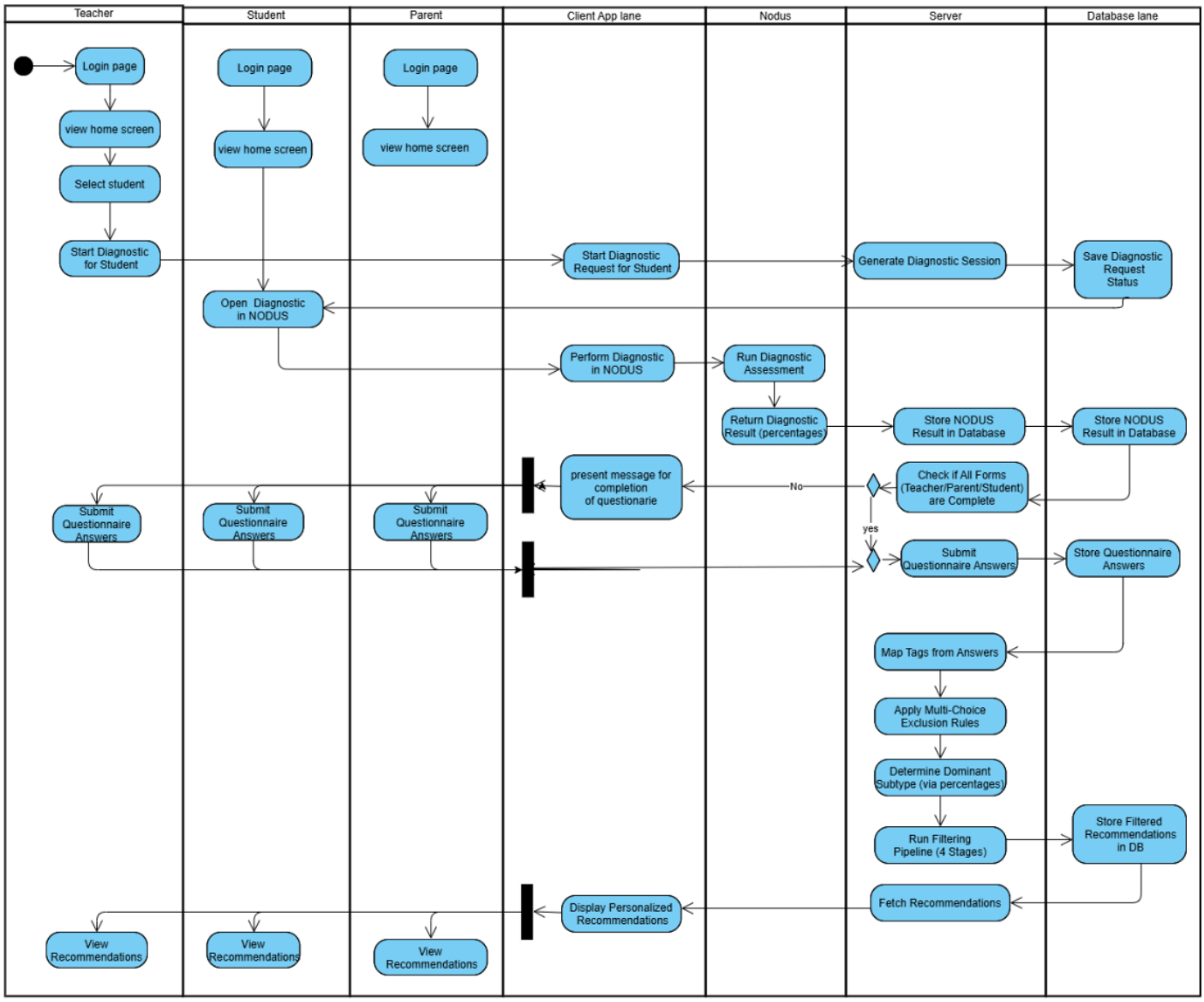
- MongoDB Database stores all system data including Recommendations, Tasks, Chat logs, Diagnostics, Answers, and User profiles
- Bidirectional data flow between the Recommendation Engine and database ensures real-time personalization

The architecture demonstrates clear separation of concerns with the frontend handling user interactions and presentation logic, while the backend manages business logic, external integrations, and data persistence. The direct connection between assessment components and NODUS enables real-time diagnostic processing, while all other data operations are centrally managed through the backend services.

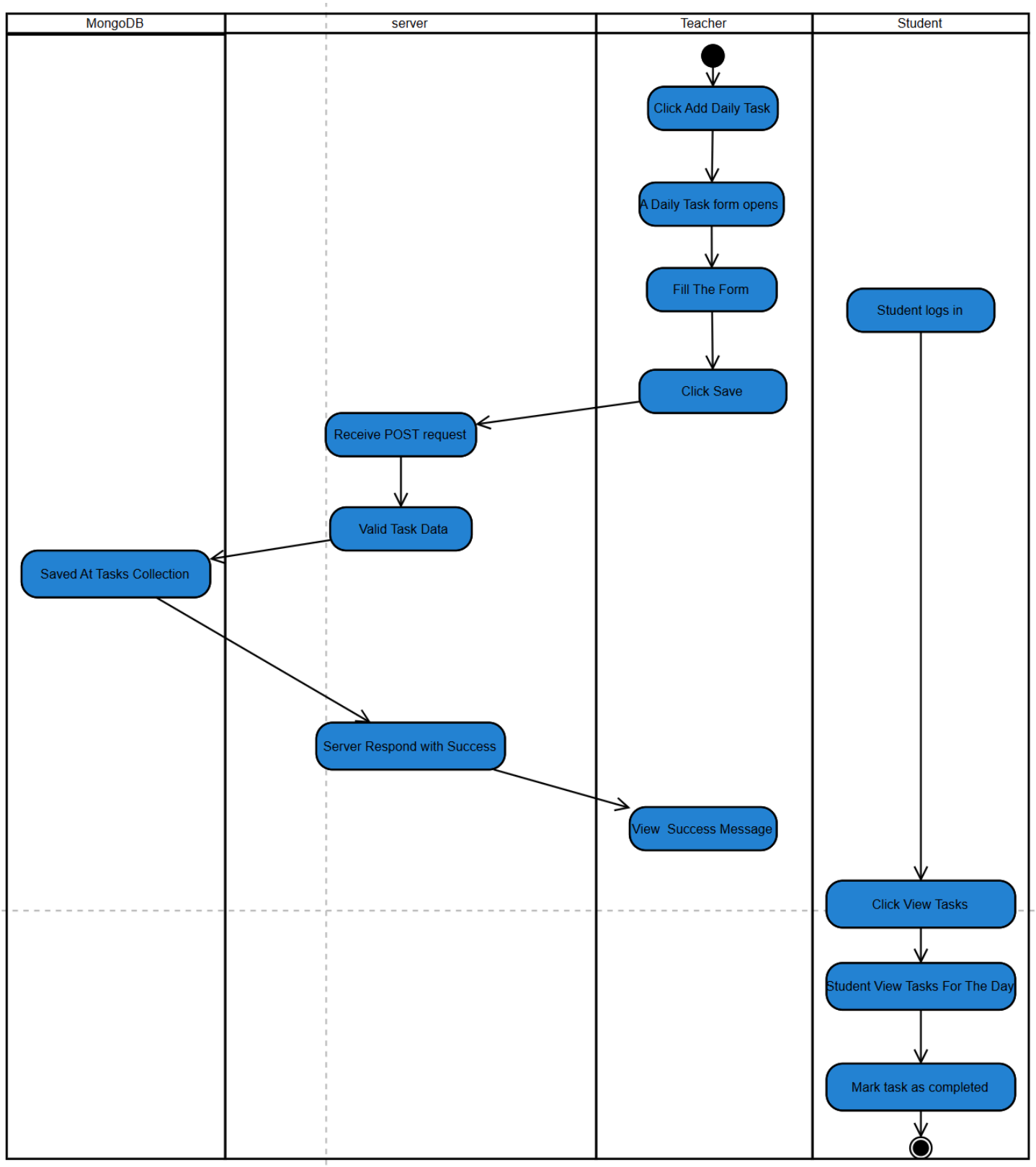
This layered approach ensures scalability, maintainability, and secure data handling across all system interactions.

4.2 System Workflow (Activity Diagram)

1. Personalized Recommendation Engine



2. Adding and Viewing a Daily Task



5. Research and Development Process

5.1 Implementation Steps

- Translated NODUS diagnostic system
- Set up the GitHub repository, branching strategy, and continuous integration via GitHub.
- Implemented the API endpoints in Express.js for diagnostic import, questionnaire submission, recommendation generation, task management, and chat messaging.
- Built React web clients, integrating UI components for questionnaires, dashboards, tasks, and chat.
- Connected to the NODUS external diagnostic service.
- Testing and Validation Process:

  1. Algorithm Testing and Validation

  Our recommendation engine required extensive validation to ensure accurate diagnostic mapping and reliable output generation:

  - Decision Tree Scenario Testing: Created comprehensive test cases covering all possible ADHD diagnostic combinations, including borderline cases where multiple subtypes might be detected simultaneously
  - Fictitious User Profile Generation: Developed detailed user personas representing edge cases such as:
    - Students with borderline threshold scores (e.g., 19% inattention, 24% hyperactivity)
    - Multiple subtype presentations requiring parent disambiguation
    - Environmental factor scenarios (recent trauma, family changes)
    - Complex allergy combinations affecting nutritional recommendations
  - Expected Outcome Validation: Each test scenario was validated against clinically expected recommendations using extensive console debugging output
  - Iterative Pipeline Refinement: Based on test results, we continuously refined the 11-stage filtering logic, adjusting thresholds, improving tag matching accuracy, and enhancing exclusion filters

  2. System Flow and Integration Testing

Comprehensive testing of all user interaction workflows to ensure seamless cross-role functionality:

- Multi-Role Workflow Testing: Validated complete teacher-parent-student interaction cycles, from initial assessment setup through recommendation delivery
- Cross-User Messaging Validation: Tested complex scenarios such as teachers messaging parents with multiple children to ensure correct message routing and student association
- Task Management Stress Testing: Evaluated system behavior under various task completion scenarios:
    - Early task completion attempts
    - Late submission handling
    - Concurrent task management across multiple students
    - Timer functionality under different system loads
- Language Switching Stability: Tested Hebrew-English language switching during active processes (mid-questionnaire, during chat sessions) to verify system stability and data persistence
- Real-Time Communication Testing: Evaluated WebSocket-based chat functionality under various network conditions, including connection drops and reconnection handling

3. Stress and Edge Case Testing

Performance and reliability testing under challenging conditions:

- Multi-User Concurrent Access: Simulated multiple teachers, parents, and students accessing the system simultaneously to identify potential race conditions or performance bottlenecks
- Data Integrity Testing: Validated system behavior during simultaneous form submissions, ensuring no data corruption or loss during concurrent operations
- Large Dataset Performance: Tested recommendation engine performance with expanded recommendation datasets and complex filtering scenarios
- Error Recovery Testing: Evaluated system recovery from various failure scenarios, including database connection issues and external API (NODUS) unavailability

4. Usability and Interface Testing

Beyond technical validation, we conducted user-centered testing:

- Cross-Browser Compatibility: Verified functionality across different browsers and devices

- Accessibility Testing: Ensured proper RTL/LTR language support and interface accessibility
- SUS Questionnaire Implementation: Conducted standardized usability testing resulting in our 83.75 ("Excellent") SUS score

This comprehensive testing approach ensures that BrainBridge delivers reliable, accurate, and user-friendly functionality across all supported workflows and user scenarios.

5.1.1 Project Management and Team Collaboration

Project Management Tools and Methodologies:

- Initial project management was conducted using Google Sheets for comprehensive tracking of:
  - Bug reports and issue resolution
  - Task assignments and ownership
  - Priority levels and dependencies
  - Progress status monitoring
- Mid-project transition to Notion for enhanced task management capabilities, providing improved collaboration features and project organization
- Established tri-weekly progress meetings to ensure alignment, discuss challenges, and coordinate development efforts between team members
- Task distribution was systematically managed with clear ownership assignment for each development component

Mentor Collaboration and Guidance:

- Regular consultation sessions with project mentor Dr. Natali Levi-Soskin following predefined timeline milestones
- Ongoing progress updates and timeline adjustments based on development realities and mentor feedback
- Technical consultation sessions for implementation decisions, including architecture choices, feature prioritization, and development best practices
- Mentor-guided emphasis on critical system components, identifying essential features versus optional enhancements to ensure project success within time constraints

5.2 Tools and Technologies Used

- **Frontend**: The frontend of our system is developed using **React** for building the user interface and TypeScript for type safety and maintainability.

- **Backend**: The backend was developed using **Node.js** and **Express.js**, with all code written in TypeScript for improved structure and maintainability
- **Database**: We used MongoDB as our main database, together with Mongoose ODM for schema definition and data modeling.
- **DevOps:**
  The project is deployed on a Hetzner VPS using **tmux** and **Nginx**, with code pulled manually from **GitHub**.
  No use in tools like GitHub Actions or Docker were used in the final deployment.

5.3 Stakeholder Communication

- Invited familiar students to test the translated NODUS diagnostic within BrainBridge after each major development iteration, gathering usability feedback and assessing alignment of imported results with the system's recommendations.
- Consulted with parents to review initial feature prototypes, refining UI components and data displays for readability, ease of use, and visual comfort.
- Presented the teacher dashboard to familiar educators, demonstrating workflow scenarios and soliciting feedback to enhance task assignment, behavior reporting, and chat functionality.

5.4 Project Constraints and Mitigation Strategies

The development of BrainBridge faced several significant constraints that required strategic mitigation approaches. These constraints influenced key decisions throughout the development process and shaped the final system architecture.

Constraint Analysis and Mitigation Approaches

Constraint Type	Description	Impact on Project	Mitigation Strategy
Time Constraint	In Phase A, insufficient time was allocated to building the recommendation dataset.	Phase B resources were diverted to dataset construction instead of focusing on delivery and refinement.	Strategic Prioritization: Focused on core functionality delivery while deferring advanced features. Implemented agile sprints with clear milestone definitions to maximize development efficiency within a compressed timeline.

Technological Constraint	Dependency on the NODUS diagnostic platform - a complex, independently-developed system requiring significant time investment to modify or deeply integrate.	Required workarounds (e.g., separating hyperactivity/impulsivity manually).	API Integration Strategy: Developed robust integration layer with comprehensive error handling. Created manual disaggregation protocol in Stages 4-6 of recommendation pipeline to separate hyperactivity and impulsivity constructs.
Scope Constraint	Attempted to implement a broad set of features (chat, timers, PDF export) alongside the core recommendation engine.	Increased workload and complexity relative to the limited timeline.	Feature Prioritization Matrix: Categorized features as core/deferred/excluded based on user impact and development complexity. Maintained modular architecture to enable future extension of deferred features.

Strategic Decision Framework

These constraints led to the adoption of a strategic decision framework prioritizing:

- Core Functionality Completeness: Ensuring the 11-stage recommendation engine and role-specific workflows were fully functional
- Technical Debt Management: Building extensible architecture to accommodate future feature additions
- User Experience Consistency: Maintaining high usability standards across implemented features rather than delivering incomplete functionality

6. Implementation Challenges and Technical Solutions

6.1 System Logic and Algorithm Challenges

The development of BrainBridge's recommendation engine presented several complex algorithmic challenges requiring careful design solutions:

- **Multi-Role Data Integration Challenge:** Unifying questionnaire responses from three different user roles (teacher, parent, student) into a coherent recommendation logic required developing

a sophisticated tag-matching system. The solution involved creating a normalized scoring mechanism that weighted responses across roles while maintaining individual perspective integrity.

- **ADHD Subtype Disaggregation:** The NODUS system outputs hyperactivity and impulsivity as a merged construct, but clinical literature suggests these should be treated as distinct factors. We addressed this through a parent-based disambiguation protocol (Stages 4-6) that allows explicit classification while preserving diagnostic accuracy.
- **Environmental vs. Neurodevelopmental Distinction:** Distinguishing between situational ADHD-like symptoms and neurodevelopmental ADHD required implementing Stage 7 (Environmental Onset Screening). This ensures that recent trauma or environmental changes trigger supplemental professional guidance without invalidating ADHD-specific recommendations.

6.2 Technical Implementation Issues

- **Role-Based Authentication and Privacy:** Implementing secure user validation without storing sensitive data (national IDs) required developing a phone/email-based verification system with pre-registered class lists. This approach maintains privacy while ensuring proper role assignment.

- **Real-Time Communication Architecture:** The teacher-parent chat system required WebSocket implementation with proper message routing and persistence. The challenge involved ensuring messages reached the correct recipients while maintaining conversation threading and read status tracking.

- **Cross-Platform State Management**: Managing complex asynchronous flows across multiple user interfaces required implementing Redux with middleware for predictable state management. This prevented race conditions while maintaining UI responsiveness during recommendation generation.

6.3 Performance Optimization and System Validation

- **Recommendation Caching Strategy**: The 11-stage pipeline required optimization to prevent unnecessary recomputation. We implemented a caching system that tracks form timestamps and diagnostic updates, reducing average response time by 14% while maintaining data freshness.

- **Database Query Optimization**: MongoDB indexing and query optimization ensured consistent response times even with complex recommendation filtering. The modular data model supports efficient querying across multiple collections while maintaining data integrity.

- **Comprehensive Testing Framework**: Validation included decision tree scenario testing with fictitious user profiles, multi-role workflow testing, and edge case validation. The SUS questionnaire provided standardized usability validation, resulting in an 83.75 ("Excellent") score.

7. Results and Analysis

7.1 Project Deliverables Assessment

BrainBridge successfully delivered all core objectives outlined in the project scope, achieving comprehensive functionality across three primary domains:

Core System Components - Successfully Delivered:

- 11-stage evidence-based recommendation engine with ADHD subtype classification
- Cross-platform React web application with role-specific interfaces
- Real-time teacher-parent communication system with WebSocket integration
- Task management module with gamification elements and progress tracking
- Complete Hebrew-English bilingual support with RTL/LTR architecture
- NODUS diagnostic integration with automated result processing

Performance Metrics:

- System Usability Scale (SUS) score: 83.75 (classified as "Excellent")
- Landing page load time: 1.96 seconds
- Cached recommendation retrieval: 3.53 seconds
- Initial recommendation generation: 4.12 seconds
- Caching optimization: 14% improvement in subsequent load times

7.2 Implementation Gap Analysis

While **core objectives** were achieved, several planned features were strategically deferred to ensure high-quality delivery of primary functionality:

Feature Category	Planned Implementation	Actual Status	Strategic Rationale
Advanced Messaging	Bidirectional teacher-student chat with centralized inbox	Implemented only one-directional: students can send help requests, but teachers cannot receive or reply in-app	Prioritized teacher-parent communication and core recommendation engine over secondary messaging flows

Calendar Integration	Dynamic calendar with live school events	Demo implementation with hard-coded events	Focused development resources on diagnostic and recommendation workflows
Analytics Dashboard	Historical progress tracking and longitudinal analysis	Deferred to future development	Concentrated on establishing stable data collection foundation
Recommendation Authoring	Advanced dataset editing and review workflows	Dataset construction completed without advanced tooling	Research-driven dataset development took precedence over authoring interface

Gap Analysis Factors:

1. **Resource Allocation**: One team member's military reserve duty effectively halved development capacity mid-project
2. **Research Requirements**: Phase A recommendation dataset construction extended into Phase B, requiring significant research investment
3. **Quality Standards**: Decision to maintain high usability and stability standards rather than delivering incomplete features

7.3 Technical Architecture Effectiveness

Architecture Success Metrics:

- **Modularity**: Clean separation between frontend components, backend services, and external integrations enables independent scaling and maintenance
- **Extensibility**: Deferred features can be integrated without architectural changes due to forward-compatible design
- **Performance**: Asynchronous pipeline processing and caching mechanisms deliver acceptable response times for educational environments
- **Reliability**: Comprehensive error handling and fallback strategies ensure system stability during external service dependencies

Linguistic Accessibility Achievement: A critical strategic success was expanding from English-only availability to complete Hebrew-English bilingual support, significantly broadening demographic accessibility within Israeli educational institutions.

To evaluate the usability of the BrainBridge platform, we conducted a standardized SUS (System Usability Scale) questionnaire with representative users (teachers, parents, and students). The

questionnaire consisted of 10 items, alternating between positive and negative statements, each rated on a 5-point Likert scale.

Responses were normalized according to the SUS scoring methodology, producing an adjusted total score. The BrainBridge platform achieved a **final SUS score of 83.75**, which is considered **"Excellent" usability** according to industry benchmarks.

Statistical Analysis:

- Mean score: 83.75 (n=4 participants)
- Standard deviation: 11.64
- 75% of participants achieved "Excellent" usability rating (≥80)
- 100% of participants exceeded "Good" threshold (≥68)
- Individual scores ranged from 67.5 to 95, indicating consistent positive reception across user types

Study Limitations: The small sample size (n=4) limits statistical generalizability. Future validation studies with larger user groups (n≥30) would strengthen confidence in these findings and provide more robust statistical power for definitive conclusions.

Implications: This result indicates that users generally found the system intuitive, well-integrated, and easy to learn, requiring minimal external support. The findings validate our design choices and confirm that BrainBridge provides a positive and effective user experience across different stakeholder roles.

A detailed breakdown of user responses and adjusted scores is provided in Appendix A.

7.4 Project Management and Development Insights

Strategic Decision Effectiveness:

- **Prioritization Framework**: The decision to prioritize core functionality over feature breadth resulted in a stable, high-quality system rather than incomplete broad functionality
- **Agile Adaptation**: Two-week sprint methodology enabled rapid iteration and regular milestone evaluation
- **Stakeholder Engagement**: Continuous testing with students, parents, and educators significantly improved usability and feature relevance

Key Development Learnings:

- **Early Dataset Development**: Future projects should prioritize data foundation establishment in initial phases rather than deferring to implementation phases

- **External Dependency Management**: NODUS integration challenges highlighted the importance of early API exploration and fallback strategy development
- **Cross-Role Testing**: Complex stakeholder workflows require systematic multi-user scenario testing to validate interaction patterns

7.5 Future Development Roadmap

Immediate Extensions (Phase C Candidates):

- **Bidirectional Messaging System**: Complete teacher-student communication with threaded conversations and notification systems
- **Advanced Analytics**: Historical diagnostic tracking, progress visualization, and longitudinal outcome measurement
- **Calendar Integration**: Live school event integration with task scheduling coordination

Technical Enhancements:

- **AI-Driven Personalization**: Natural language generation for recommendation customization based on user feedback patterns
- **Advanced Recommendation Engine**: Machine learning integration for recommendation effectiveness optimization
- **Mobile Application Development**: Native iOS/Android applications for enhanced accessibility

Platform Scalability:

- **Rich Media Integration:** Transition from text-only recommendations to multimedia-rich content delivery system supporting images, videos, and interactive elements tailored to each user role's needs
- **Content Management System:** Advanced authoring tools enabling easy creation and management of visual, audio, and interactive recommendation content
- **Multi-Institution Support**: Database sharding and multi-tenancy architecture for school district deployment
- **API Ecosystem**: Public API development for third-party educational tool integration
- **Advanced Security**: Enhanced authentication systems and data protection compliance

Research Integration:

- **Outcome Validation**: Longitudinal studies measuring recommendation effectiveness and student outcome improvements

- **Clinical Validation**: Collaboration with ADHD specialists for recommendation accuracy validation and refinement

8. User Guide

This section provides the **User Guide and Operating Instructions** for BrainBridge.
It is intended for teachers, parents, and students who use the system in its standard (“nominal”) flow.
Error handling and exceptional cases are not described here, as the focus is on typical, successful usage scenarios.

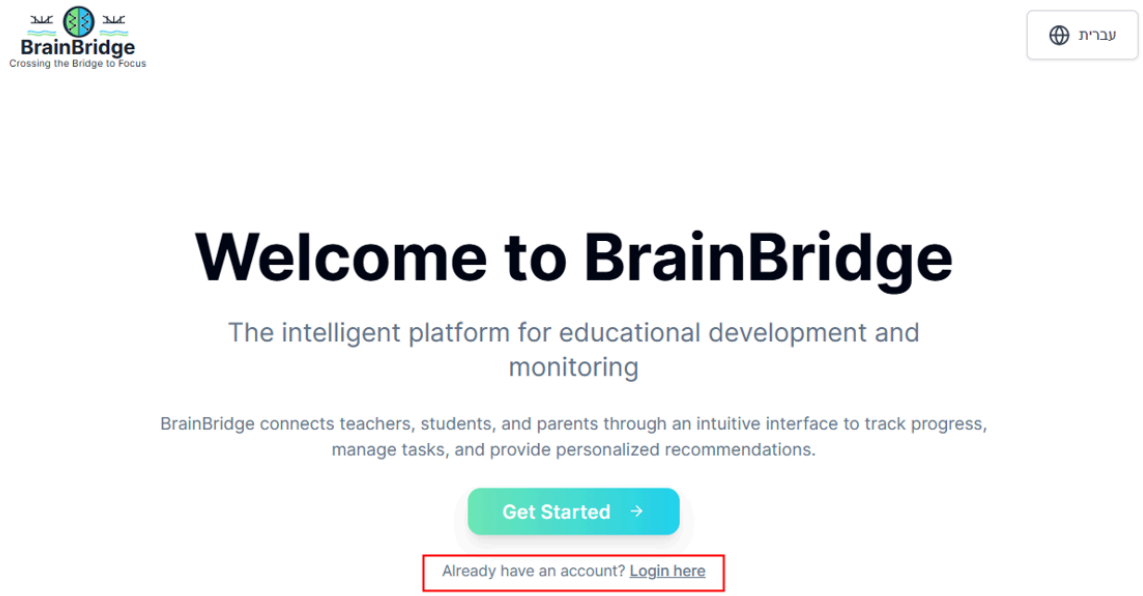
Operating Instructions (Nominal Use Cases)

8.1 Login & Registration

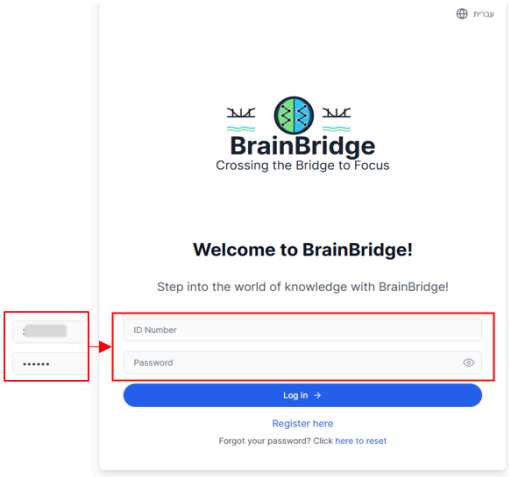
The login and registration process is identical for teachers, parents, and students. Therefore, it is described once here and applies equally to all user roles.

8.1.1 Login

- Open the BrainBridge web portal.



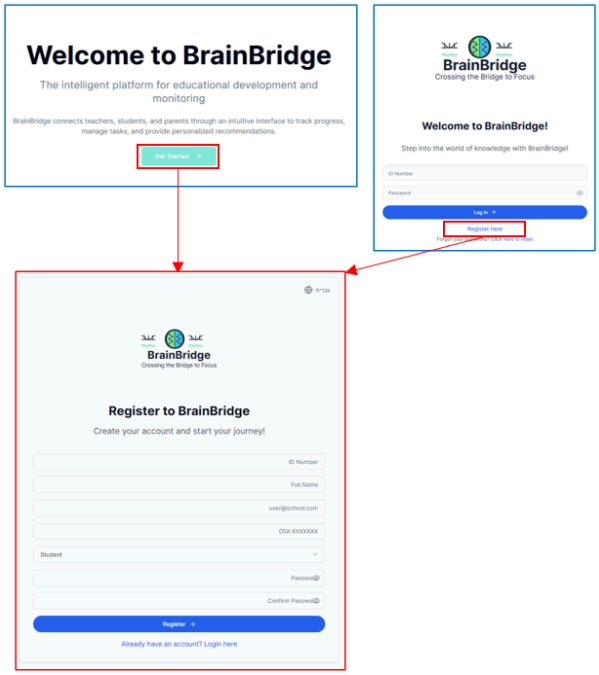
- Enter credentials and log in.



8.1.2 Registration

Important Note: BrainBridge is designed as a school-integrated system. Student registration requires pre-enrollment in the school's database. Users cannot create accounts independently and must first be added to the system by their school administrators. This ensures proper class assignment and maintains data security within the educational institution.

Open the BrainBridge web portal and search for the “Get started” or “Register here” Button.



8.2 Reset password

Purpose: allow any user (teacher/parent/student) to set a new password when they can’t log in.

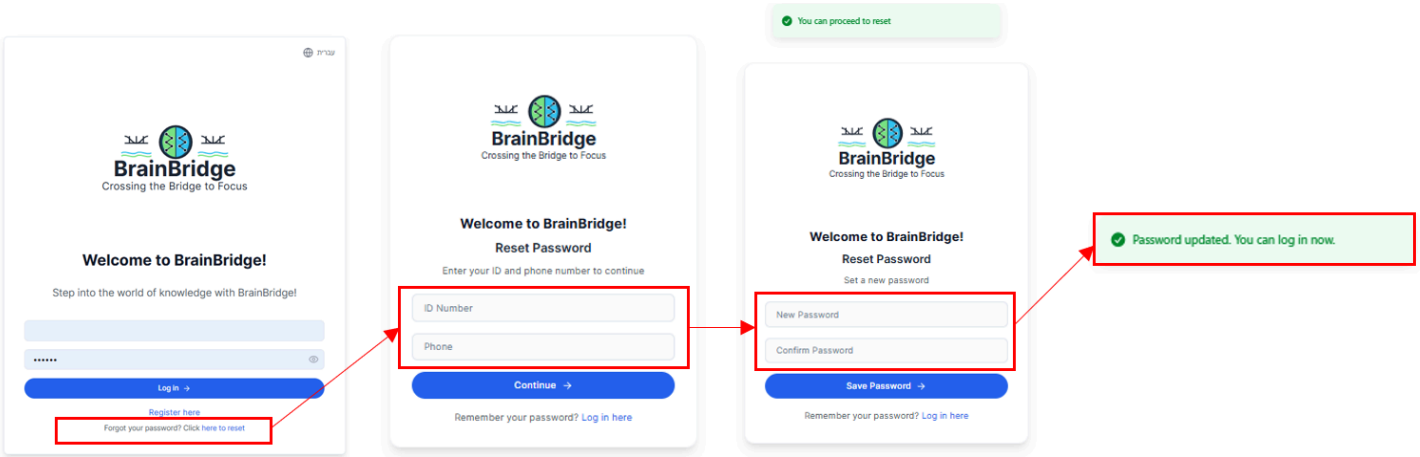
Prerequisites

- The user already exists in the school's BrainBridge roster.
- You know the **ID Number** and **Phone** recorded in the school system.

Steps

1. **Start the reset**
    - On the login page, click **"Forgot your password? Click here to reset"**.
2. **Verify your identity**
    - On the **Reset Password** screen, enter your **ID Number** and **Phone**.
        - Use digits only (include leading zero if applicable).
    - Click **Continue**.
    - If the details match the school records, a green banner appears: **"You can proceed to reset."**

      If they don't match, correct the details and try again. If the issue persists, contact your school administrator to update your information.
3. **Set a new password**
    - Enter **New Password** and **Confirm Password**.
    - The two fields must match. (Recommended: at least 8 characters and a mix of letters and numbers.)
    - Click **Save Password**.
    - On success you'll see: **"Password updated. You can log in now."**
4. **Log in**
    - Click **"Log in here"** (link on the page) or return to the login screen.
    - Sign in with your usual user identifier and your **new** password.



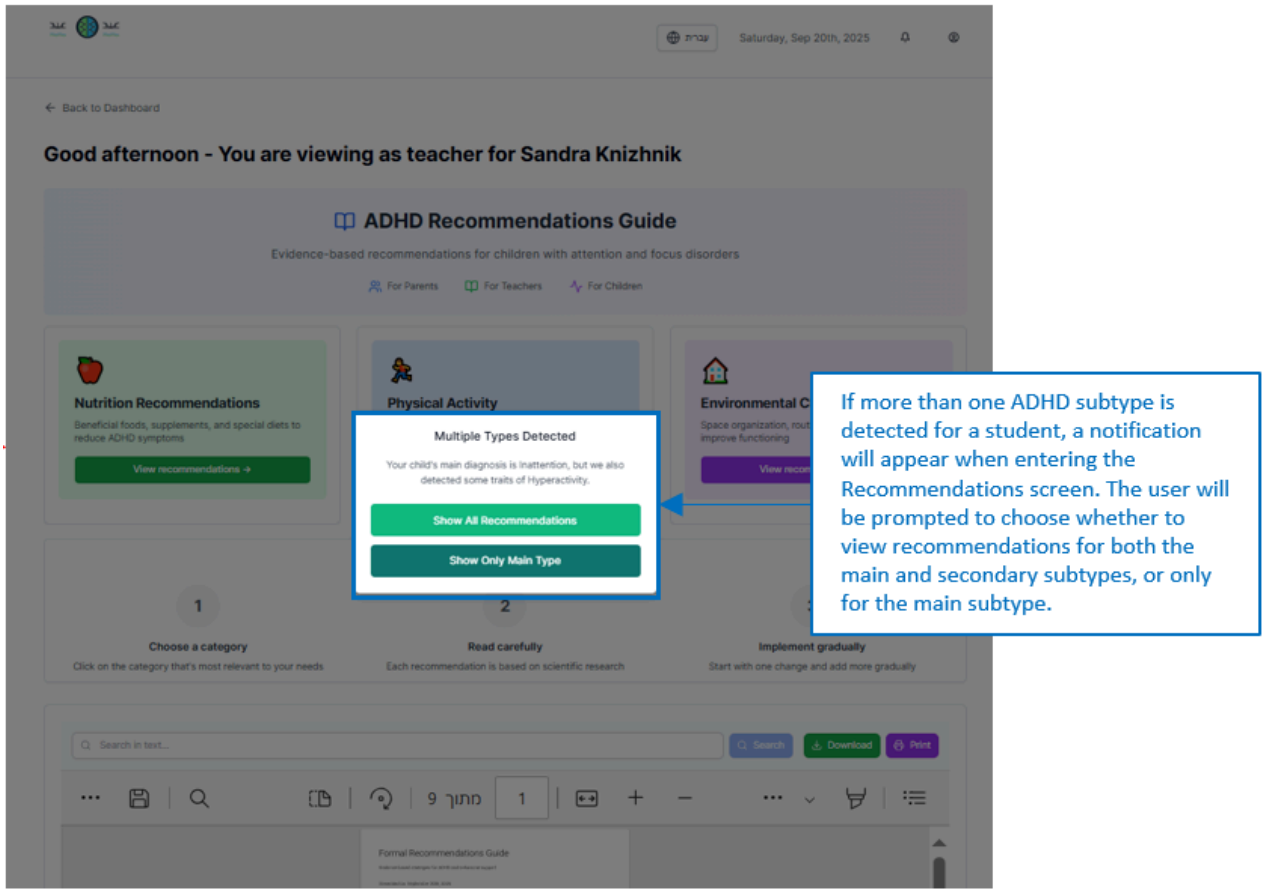
Troubleshooting

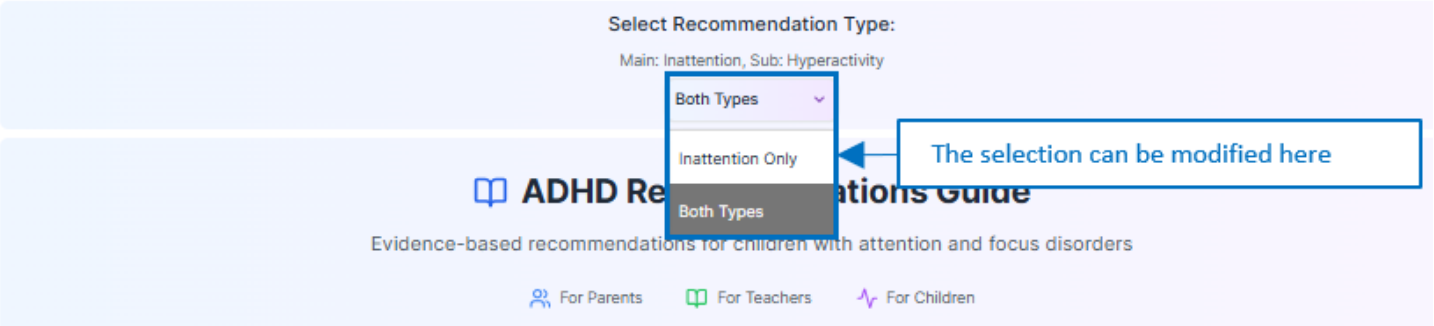
- **No match found:** Your ID/Phone may be different from what's stored. Contact the school admin to verify or update your record.
- **Validation errors:** Follow the on-screen hints (e.g., passwords must match).
- **Still locked out:** Ask the school administrator to assist or to trigger an admin-side reset.

8.3 View Recommendations

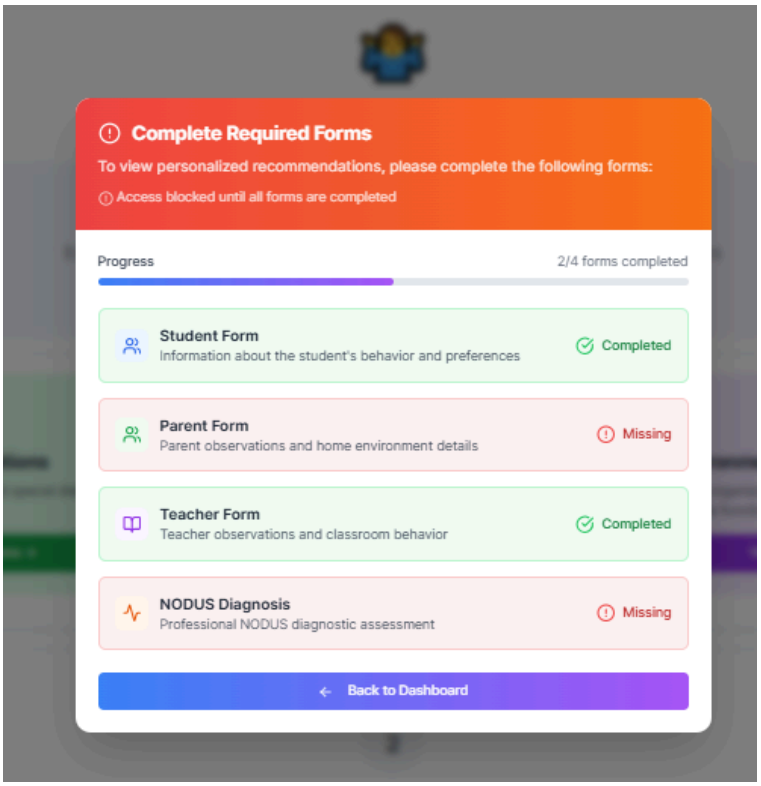
The recommendation view is identical for all users (teacher, parent, and student). Therefore, the instructions provided in the Recommendation Display Options section apply equally across roles.

When multiple ADHD subtypes are detected, the user will first receive a notification upon entering the Recommendations screen. This choice can later be adjusted in the selection menu at the top of the page.





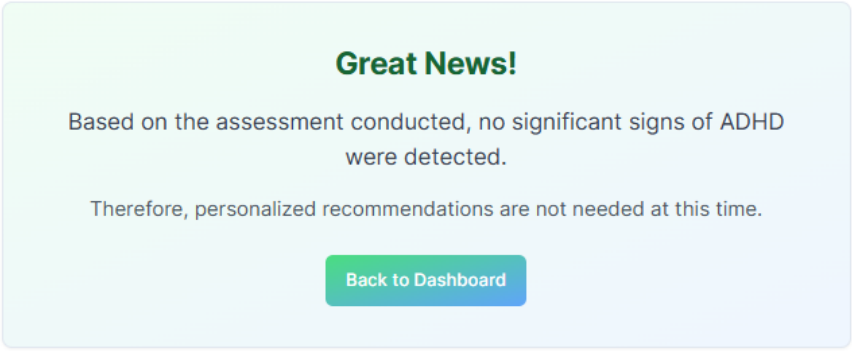
* If some information is missing (e.g., a questionnaire or diagnostic has not yet been completed), the system will display a notification indicating exactly which items are still pending before recommendations can be generated.



No ADHD Indicators Detected

If the diagnostic results indicate no significant signs of ADHD, the system will display a confirmation message instead of personalized recommendations. In such cases, the user is informed that no targeted interventions are required, and the recommendations section will remain empty.

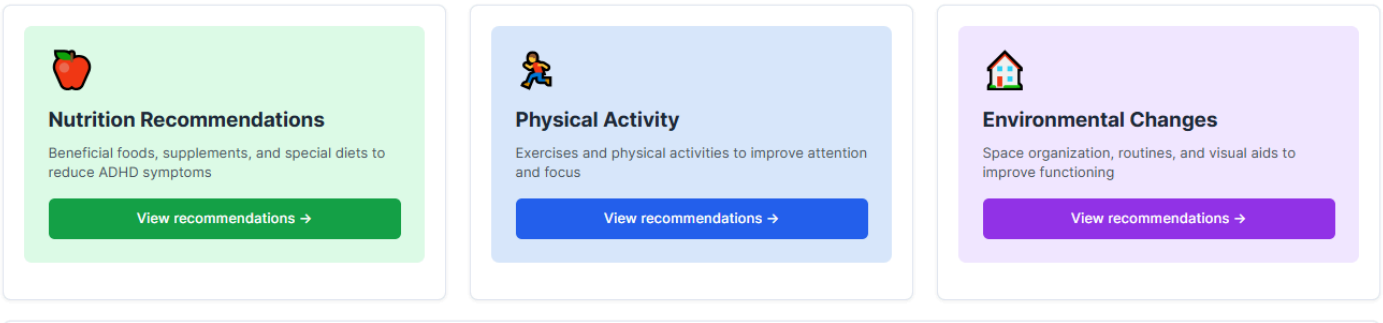
The screen will show the following message:



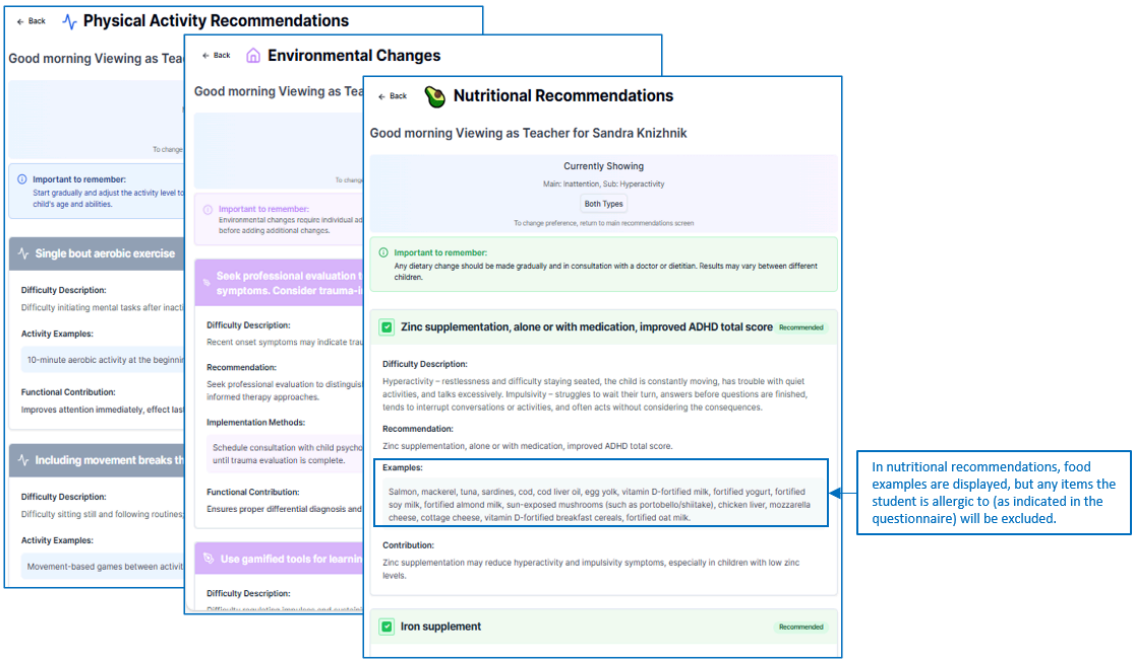
This ensures clarity for teachers, parents, and students, confirming that the diagnostic outcome does not warrant further intervention.

Recommendation Display Options - Generated Recommendations can be viewed in two formats:

1. **Tab View**
    - Each card is organized under a category (Environmental, Nutritional, Physical Activity). - Users can navigate easily between categories.



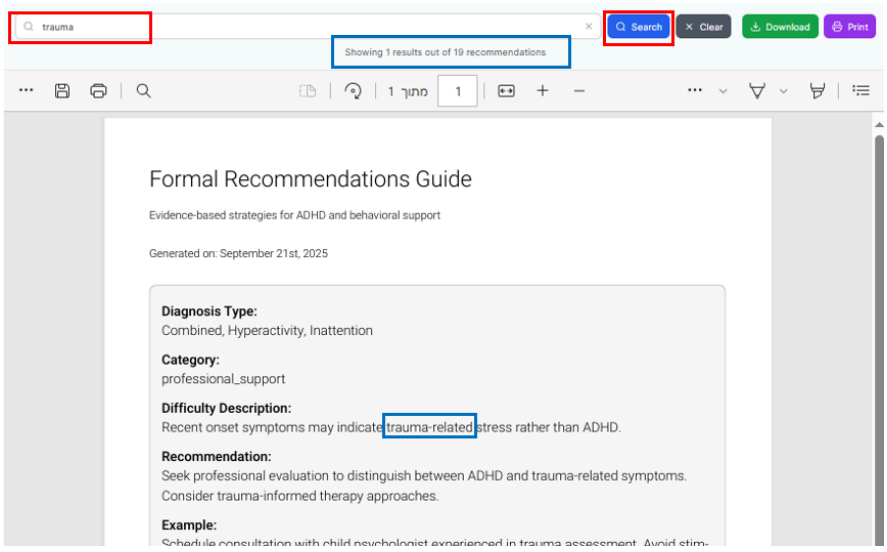
    - Recommendations are displayed as interactive cards.



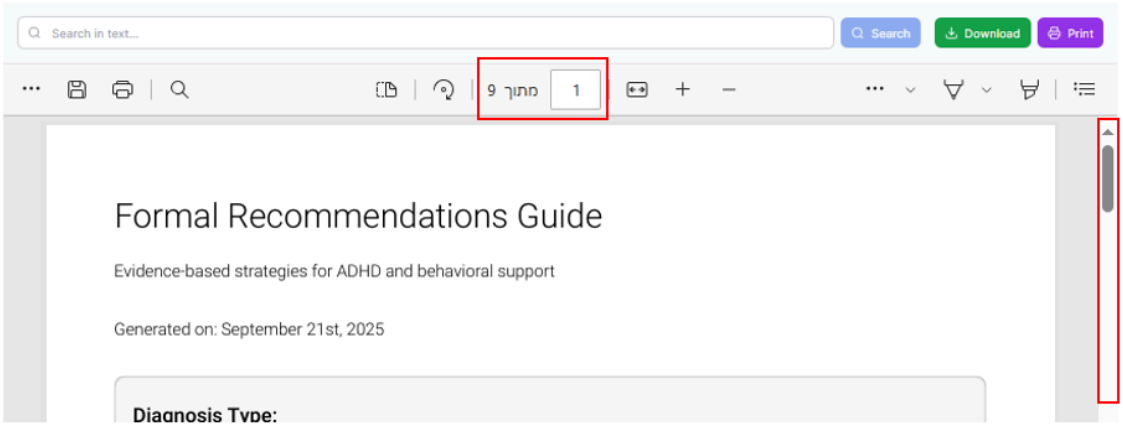
2. **PDF View - Viewing and Navigating the Recommendations PDF**

The Formal Recommendations Guide is generated as a PDF document for easy reading, searching, and sharing. Users can interact with the PDF using several built-in tools:

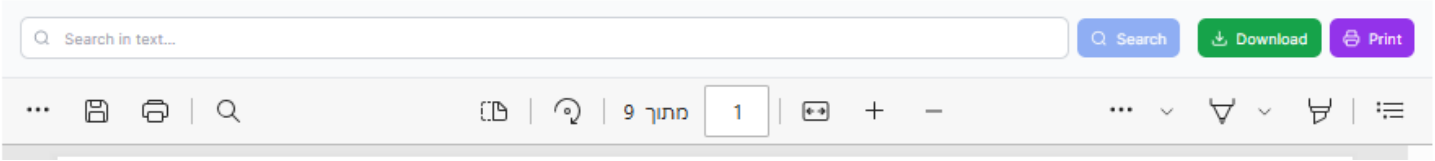
- Search Function: Enter any keyword (e.g., trauma) in the search bar to instantly filter and highlight relevant recommendations. Only the matching recommendations will remain visible, making it easier to focus on specific needs.



- Navigation Controls: Use the arrows or page number selector to move between pages. The system will also display how many results were found and where they appear.



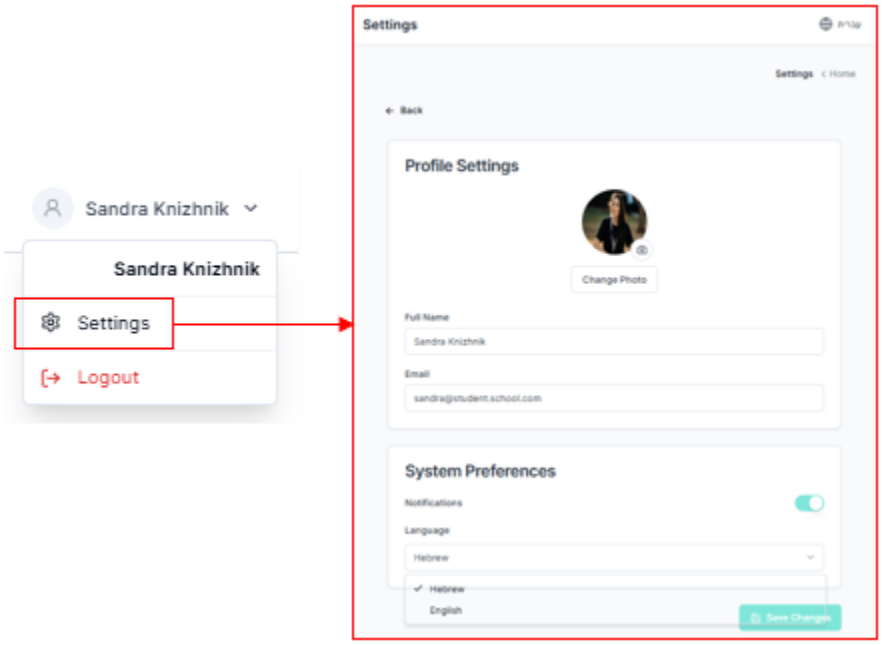
- Download & Print: At any time, the full recommendations file can be downloaded or printed for offline reference.



This functionality ensures that long recommendation lists remain accessible, easy to navigate, and tailored to the user's immediate search needs.

8.4 User Profile & System Preferences

Every user in BrainBridge has access to a personal Profile & Settings page, which looks identical across all user roles (Teacher, Parent, and Student).

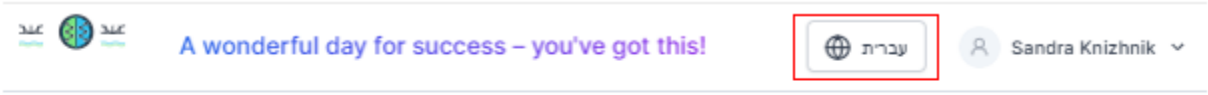


From this page, users can:

- **Update their profile information** - including name, email, and profile picture.
- **Manage system preferences** - such as enabling/disabling notifications.
- **Select interface language -** switch between Hebrew and English.

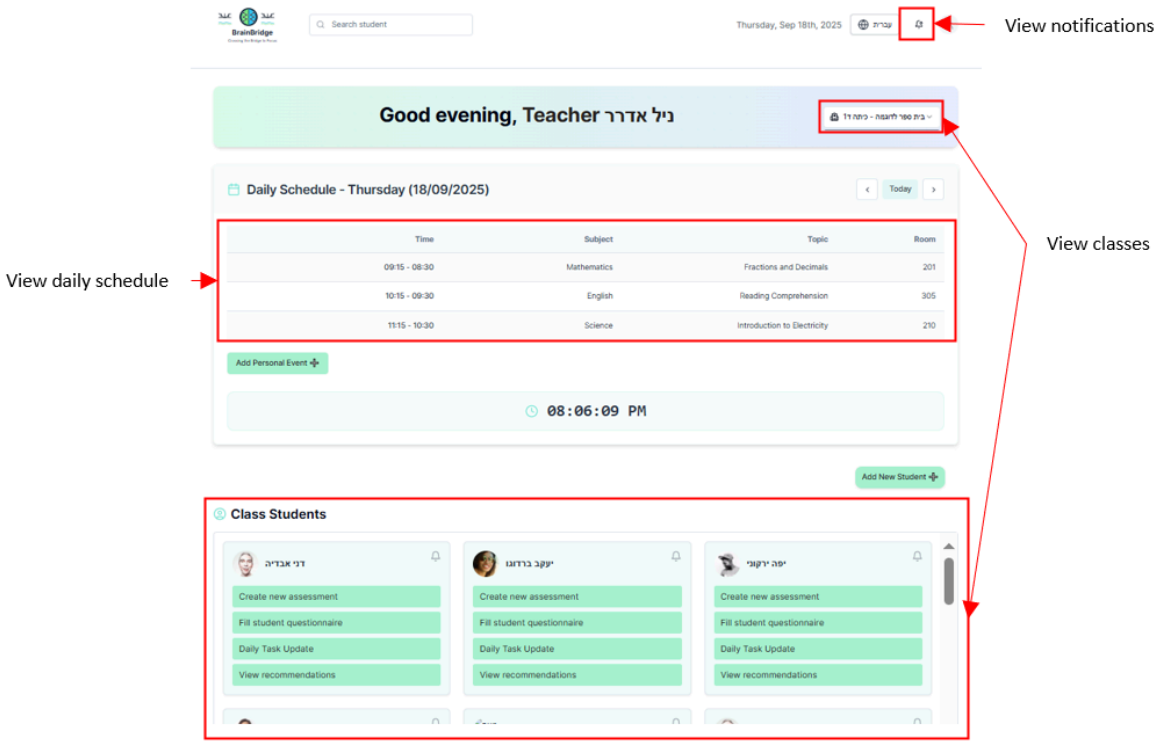
All settings are stored securely and applied immediately across the system once the user saves changes.

In addition to the Profile page, the **language can also be changed directly from the top navigation bar**, allowing quick switching between Hebrew and English on any screen.

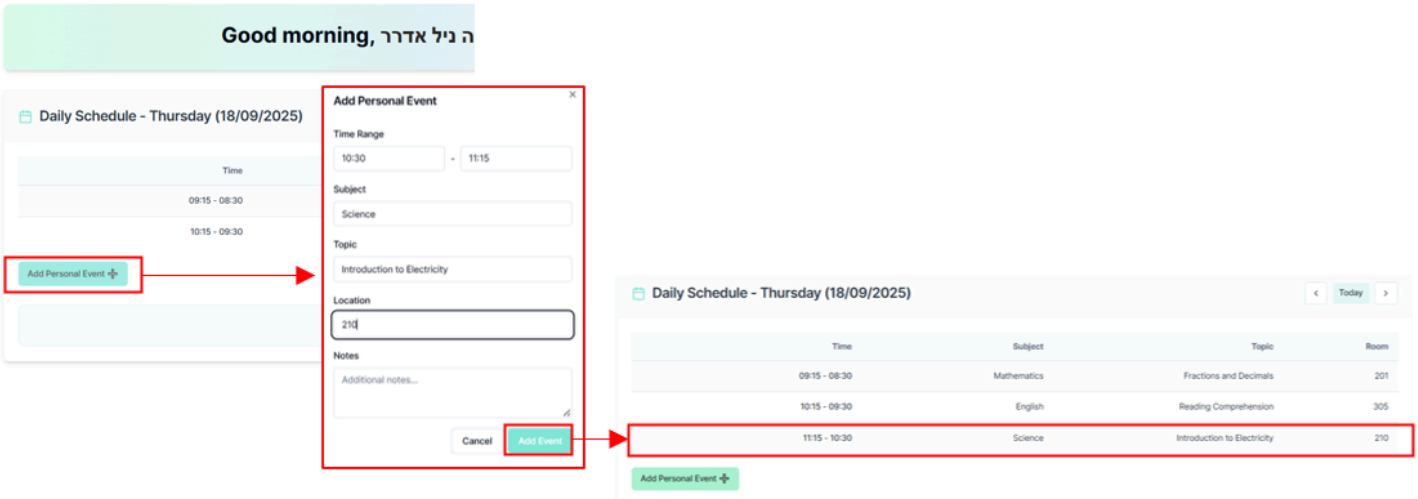


8.5 Teacher Workflow

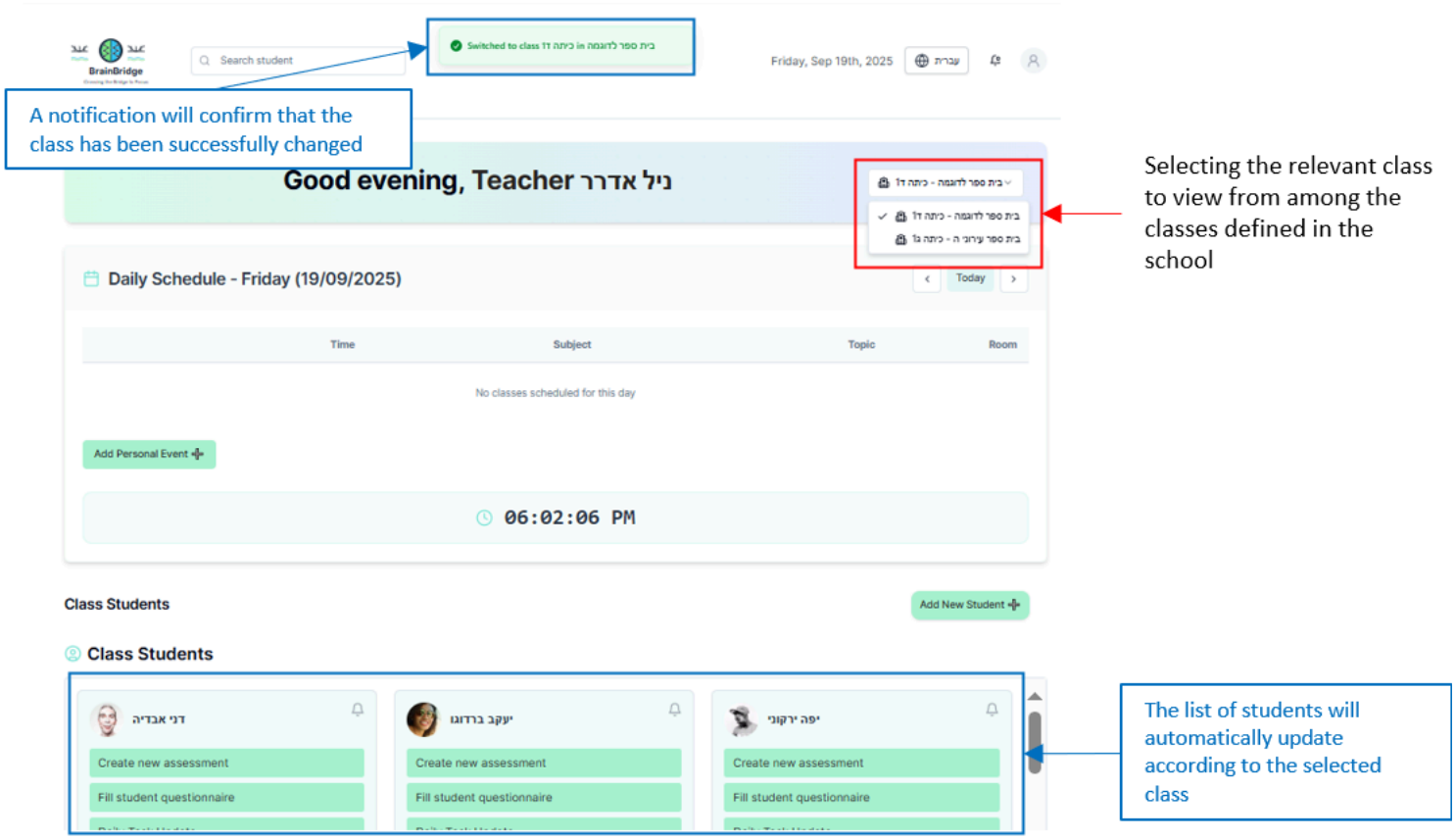
1. Access Dashboard
    - After login, you will be redirected to your teacher dashboard where you can view classes, daily tasks, and notifications.



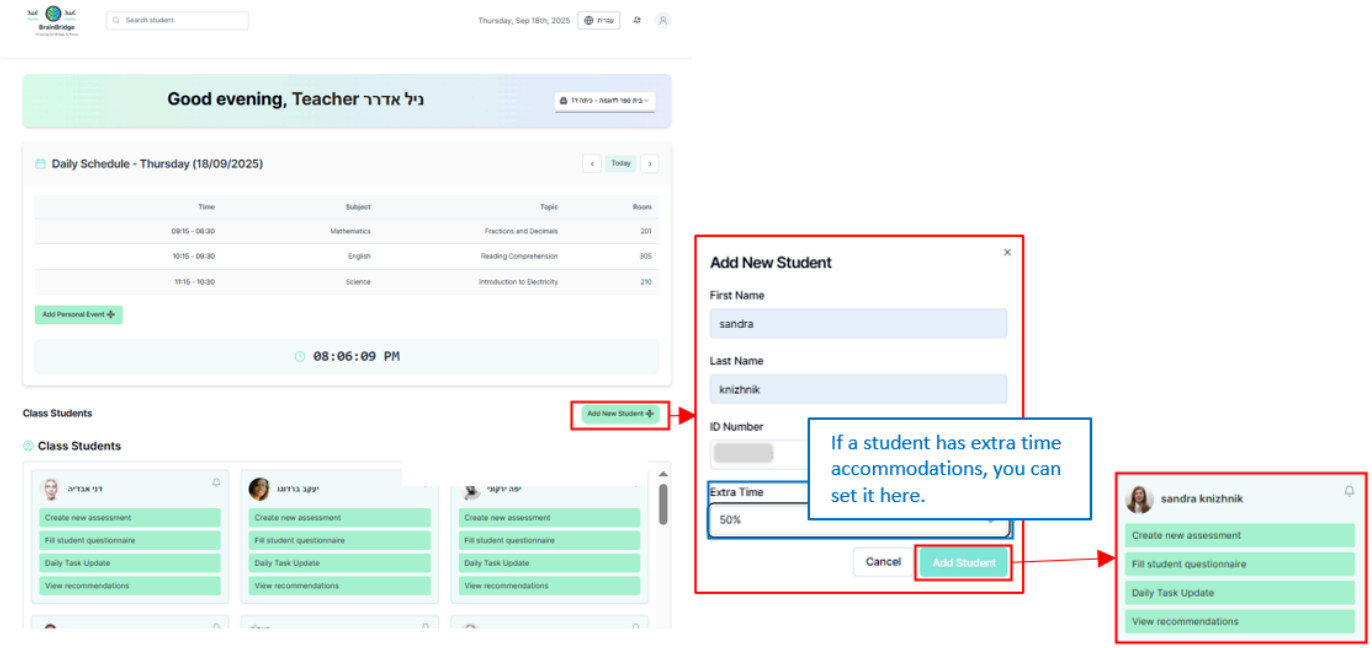
    - Add a personal task to schedule



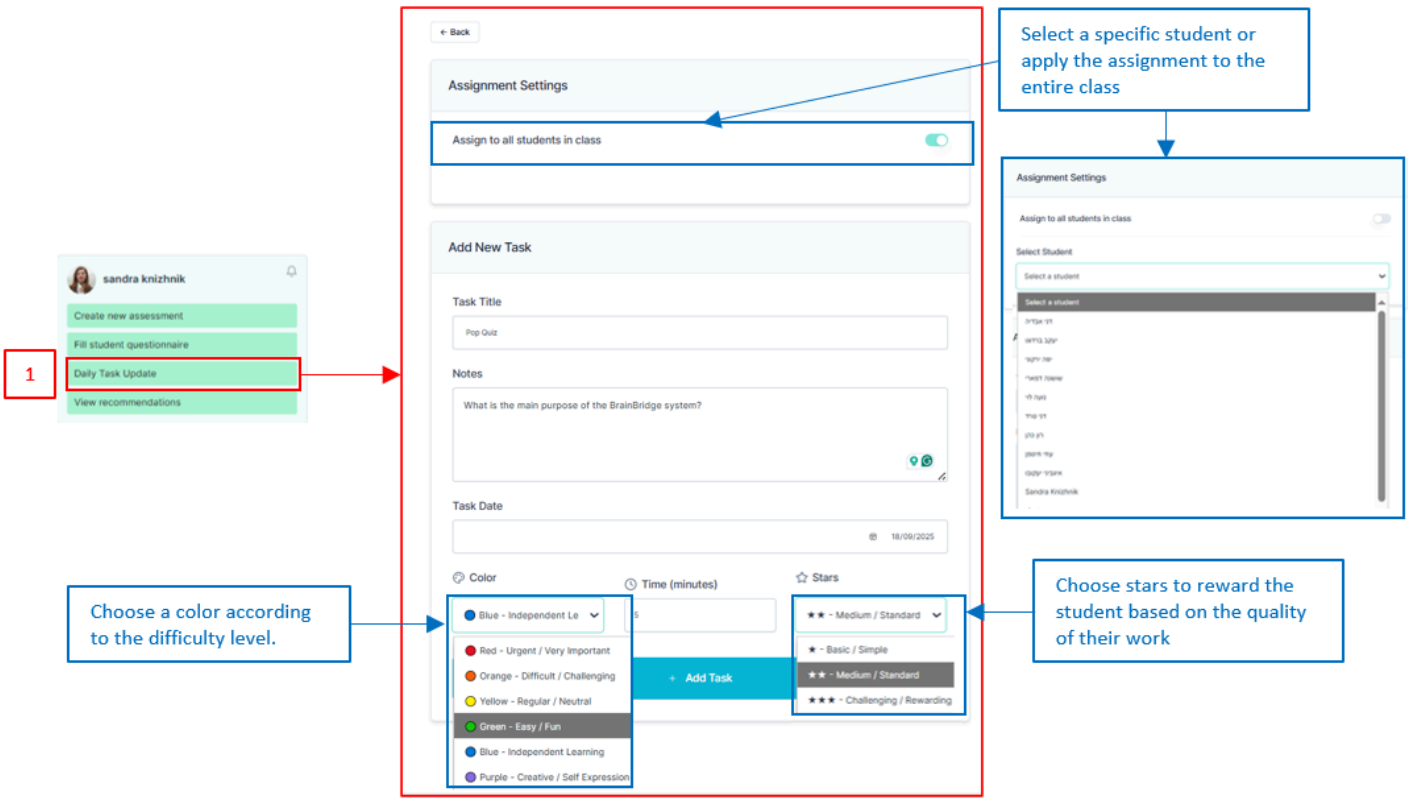
2. Select a Class and Student
    - Choose the relevant class.



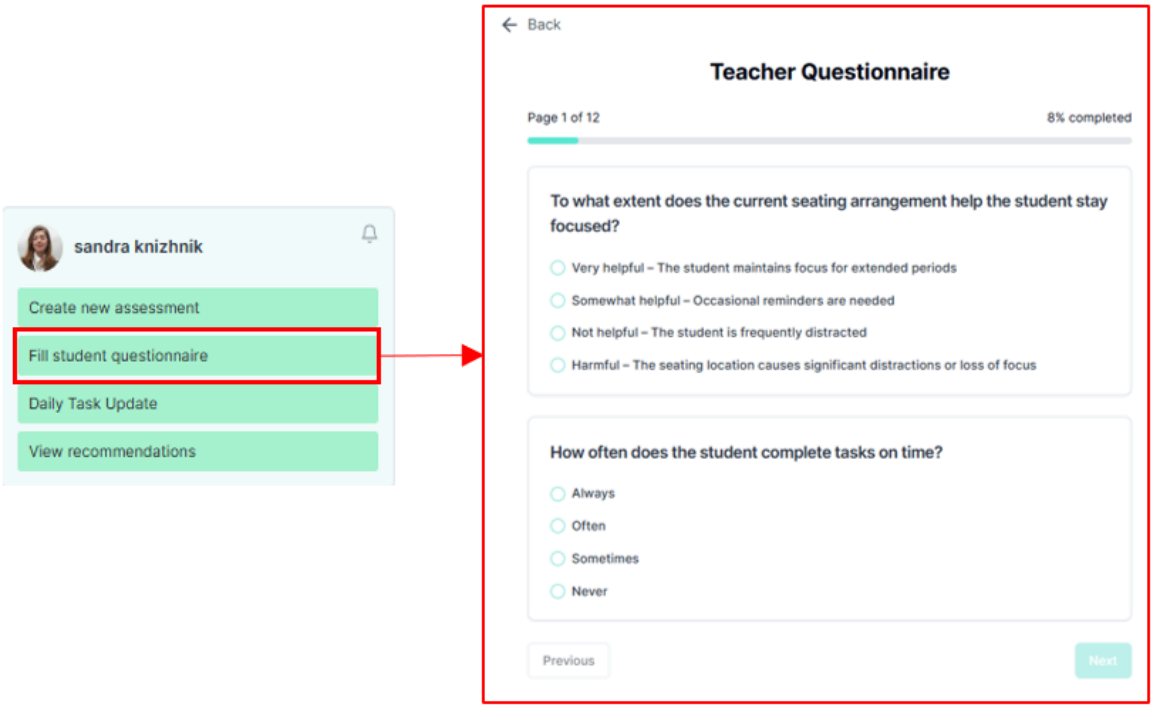
- Add a student to class



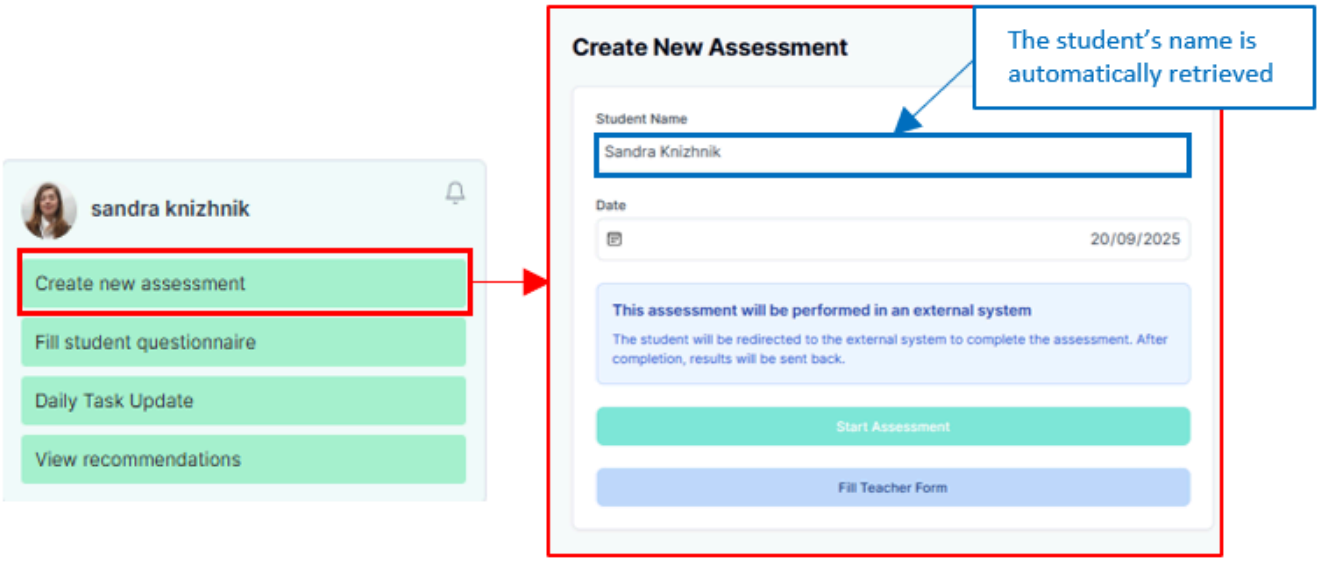
3. Assign Tasks and Questionnaires
   - Create daily tasks such as academic exercises or behavioral activities.



- Fill out questionnaires to students.

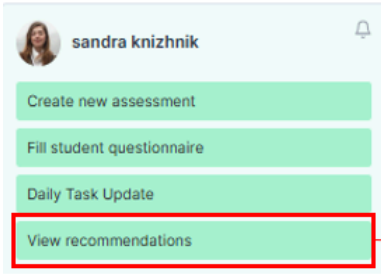


4. Trigger Diagnostic Process (if required)
   - If a diagnostic assessment is needed, initiate it from the dashboard.

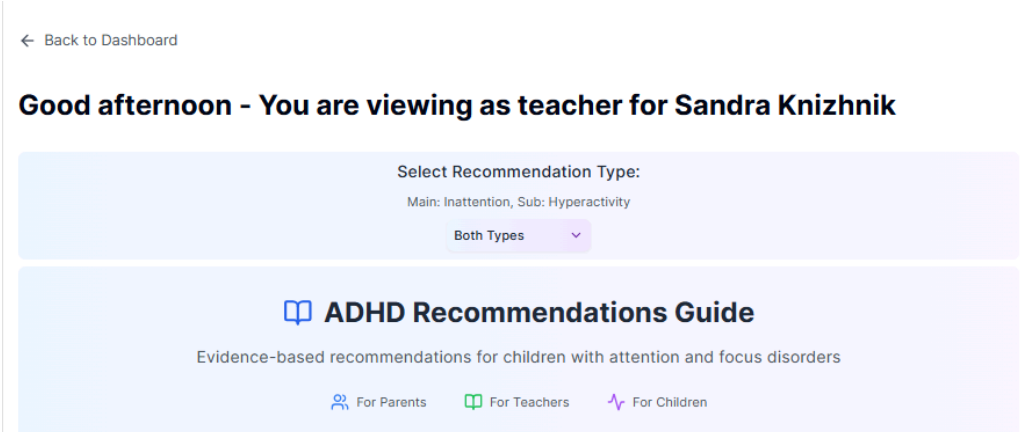


* When the teacher initiates a diagnostic process for a student, a Start Diagnostic button becomes available on the student's dashboard. By clicking this button, the student is redirected to the external NODUS system to complete the assessment

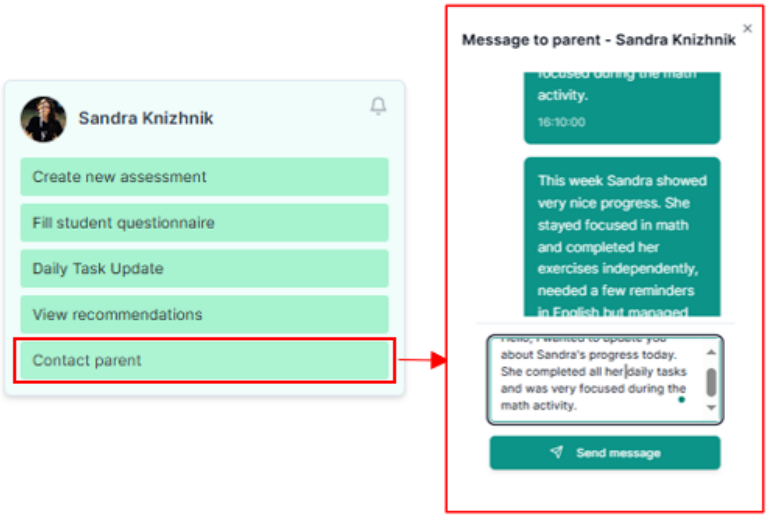
5. View Recommendations
    - Once all questionnaires and diagnostics are complete, click **View Recommendations.**



* The recommendation view is identical for all users (teacher, parent, and student). Therefore, the instructions provided in the Recommendation Display Options section apply equally across roles.



6. Communicate with Parents
    - Use the built-in chat module to share progress, updates, and personalized strategies with parents.

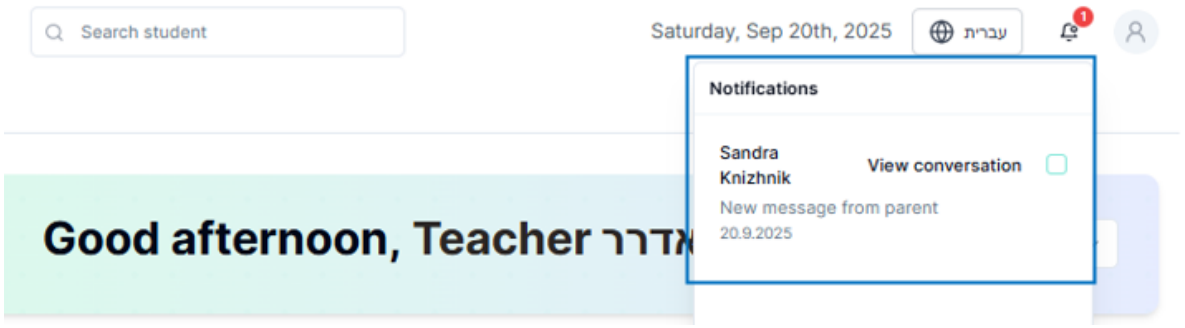


Note:

The *Contact Parent* button will only appear if a parent is assigned to the selected student in the school's database.

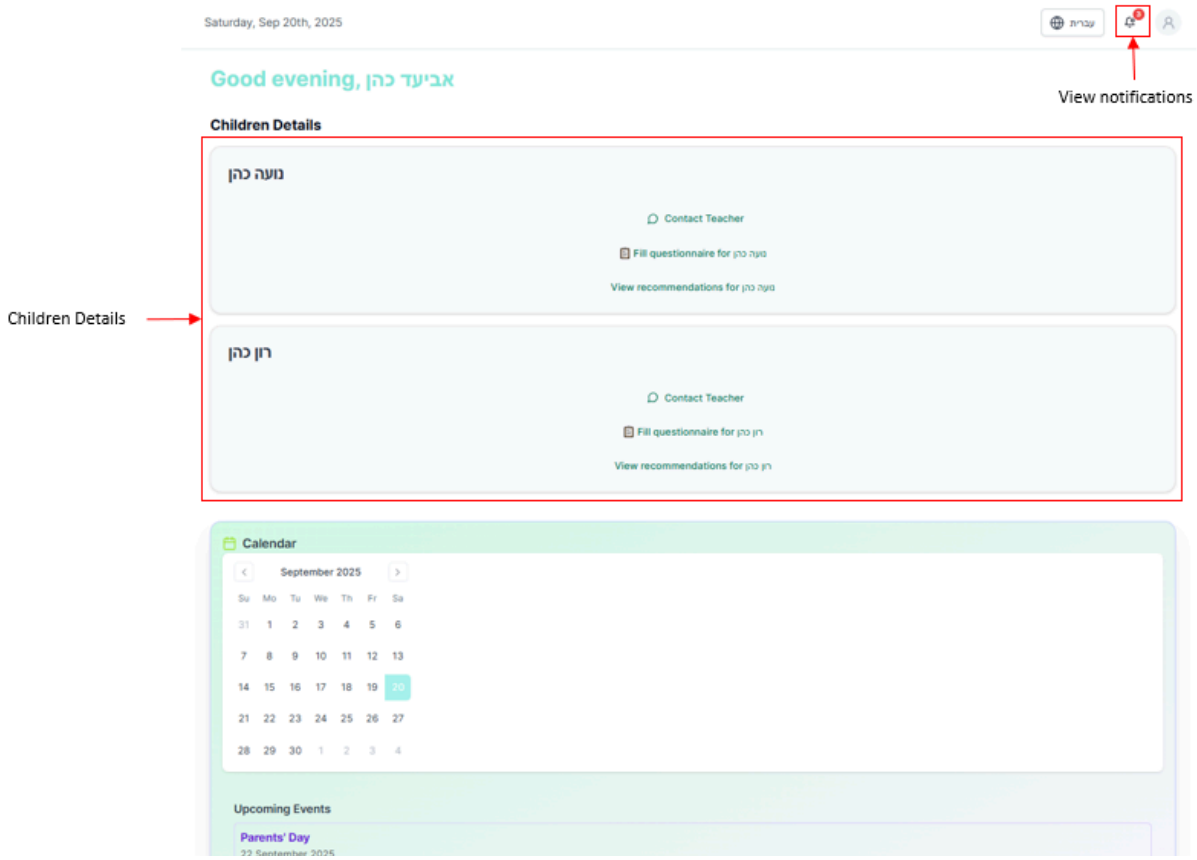
If no parent record exists, this option will not be visible in the student's card.

    - Notifications can be viewed from the toolbar.



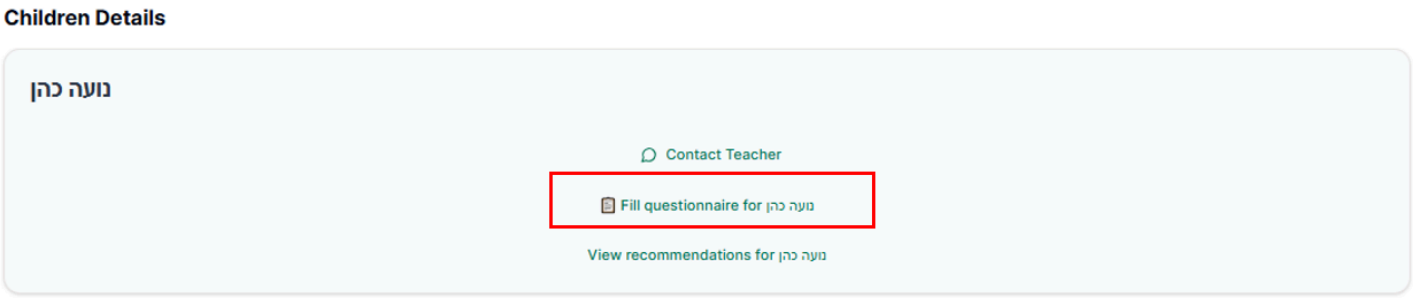
8.6 Parent Workflow

1. Access Dashboard
    - After *login*, you will be redirected to your parent dashboard where you can view your child's profile, view recommendations, notifications, chat with teacher.



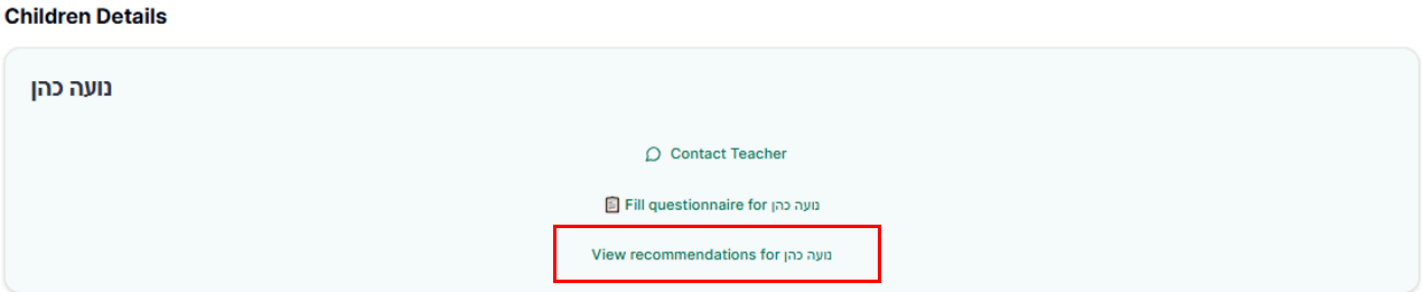
2. Complete Parent Questionnaire
    - Fill in the parent questionnaire.

* Your responses are stored securely and used to generate personalized recommendations



3. View Recommendations

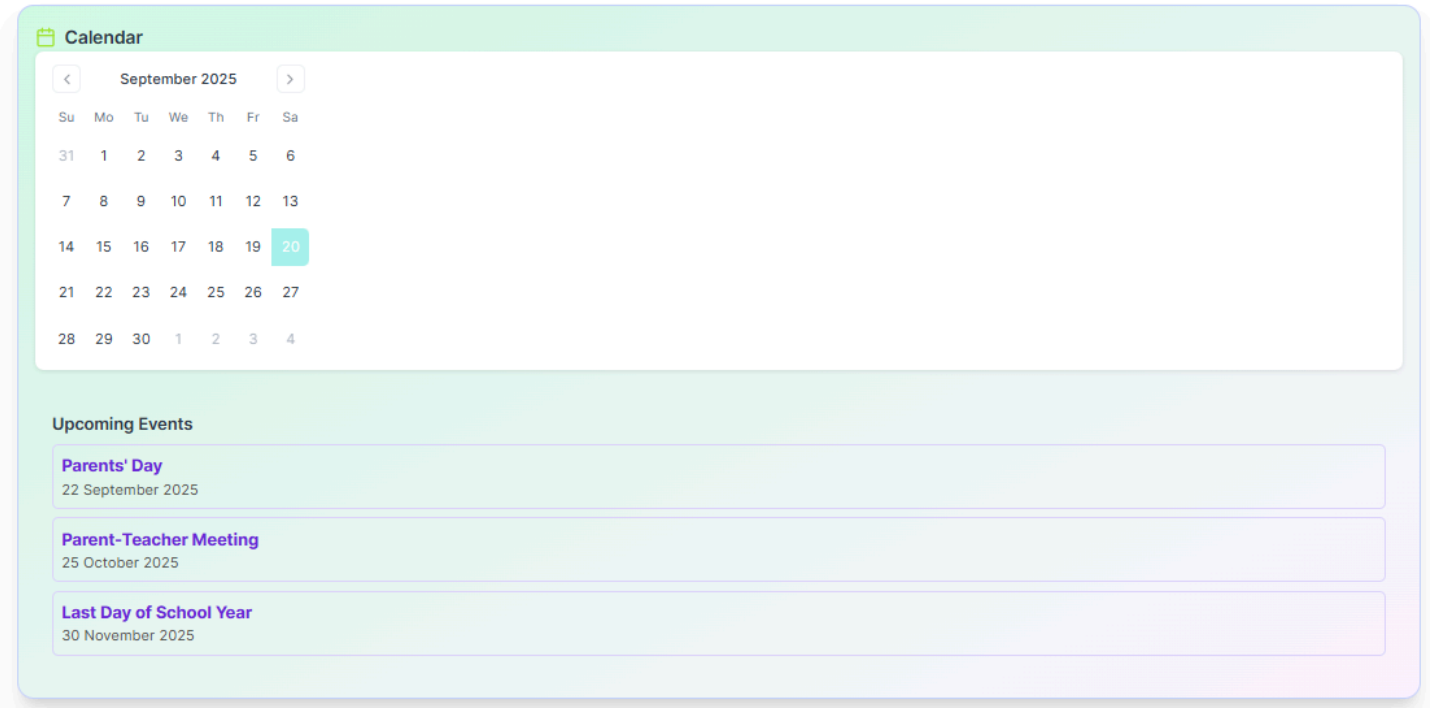
* Once the diagnostic and questionnaires are complete, recommendations will be displayed



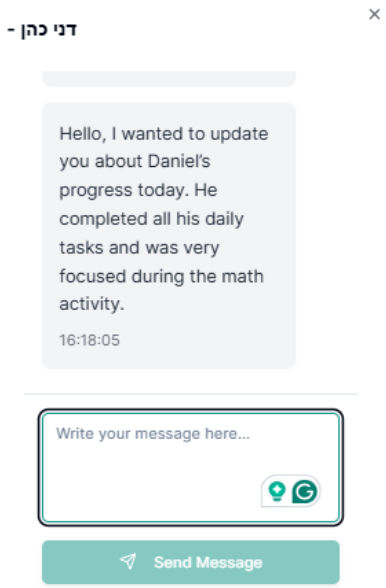
- As noted earlier, the recommendation view is identical across all roles and follows the same instructions provided in the *Recommendation Display Options* section.

4. 🔹 **Demo Feature -** View Upcoming Events
   - Parents can also view upcoming events directly from the calendar section, including Parents' Day, Parent-Teacher Meetings, and the Last Day of the School Year

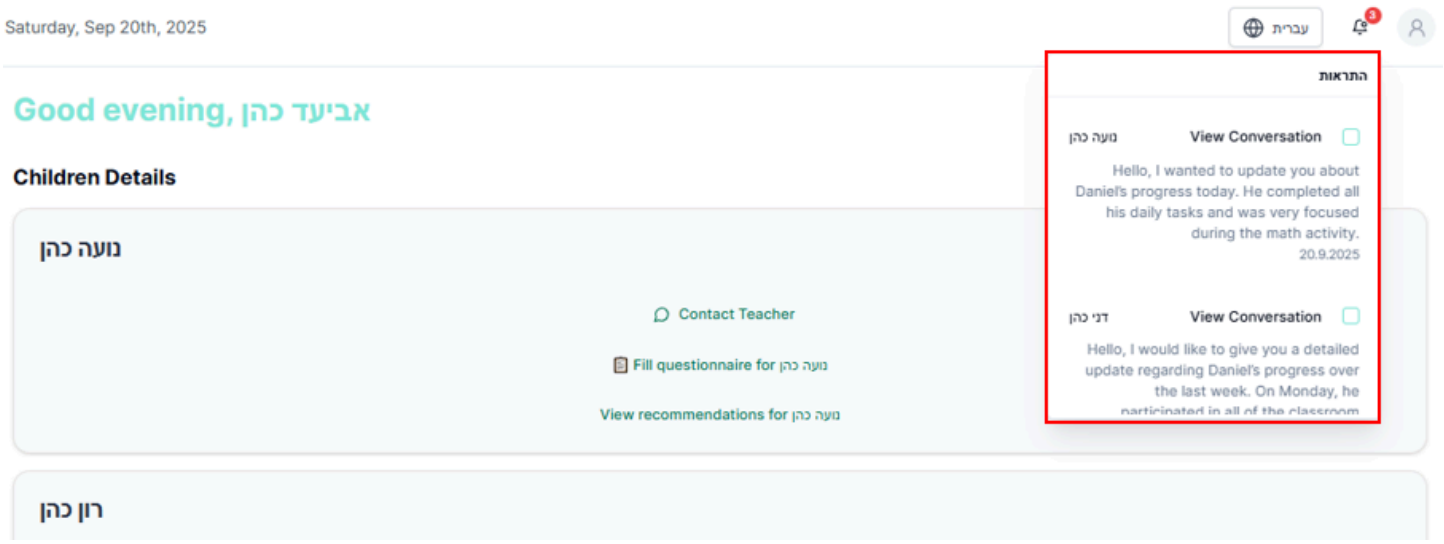
   The *Upcoming Events* section is currently for demonstration only. The events are hard-coded and do not reflect live data.



5. Communicate with Teacher
    - Use the integrated chat module to share observations from home or to ask questions.

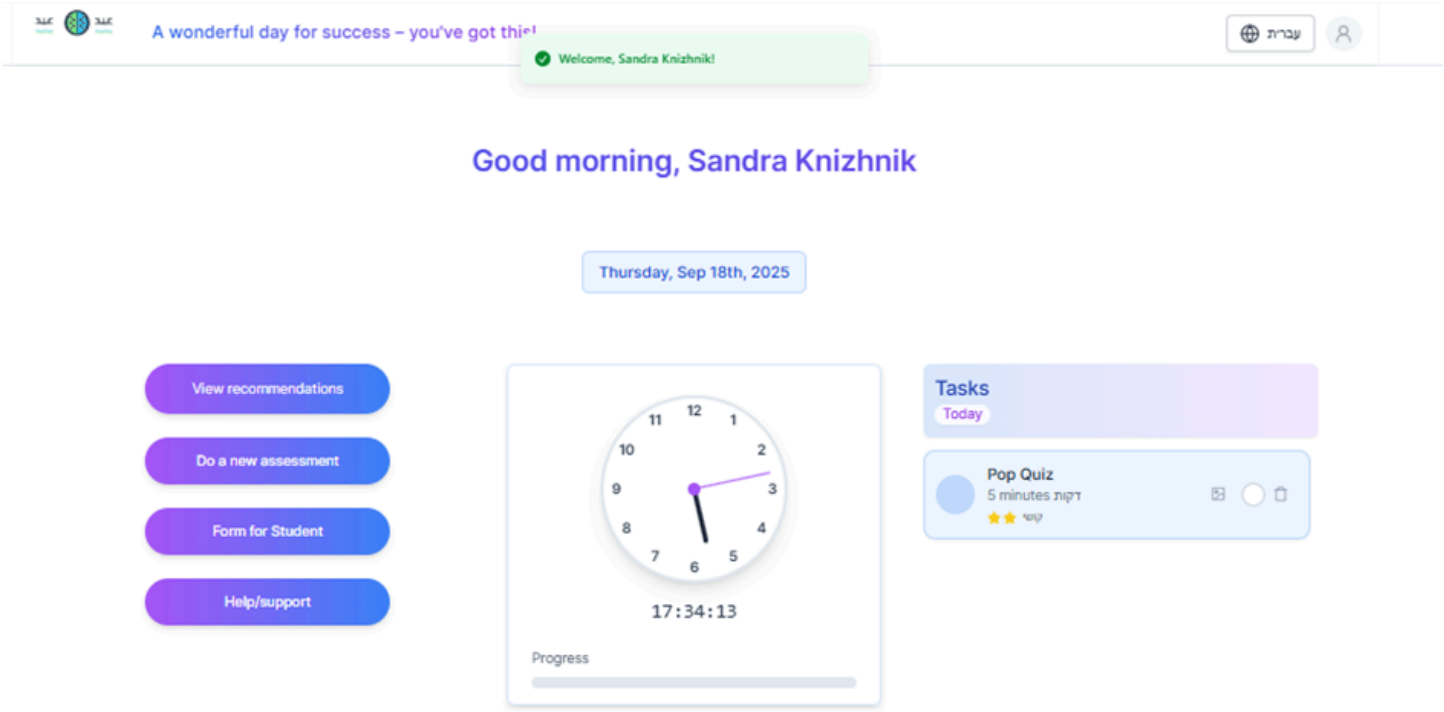


    - Notifications of new messages or updates can be accessed from the toolbar.



8.7 Student Workflow

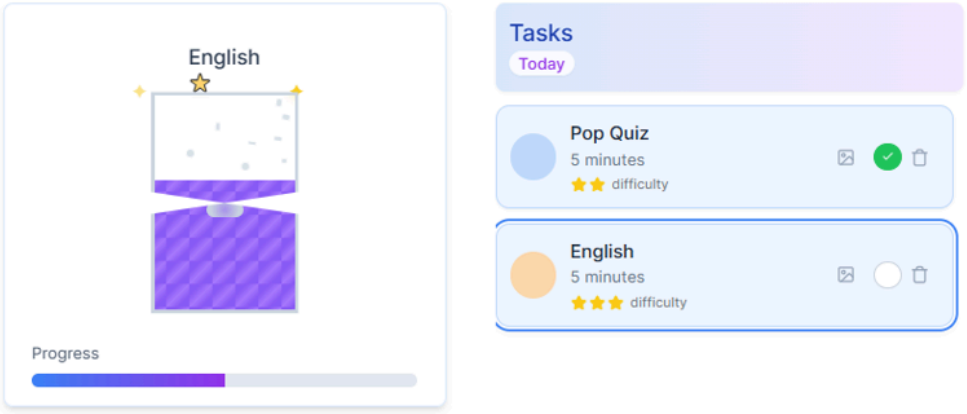
1. Access Dashboard
    - After *login*, you will be redirected to your student dashboard where you can see your daily assigned tasks and progress.



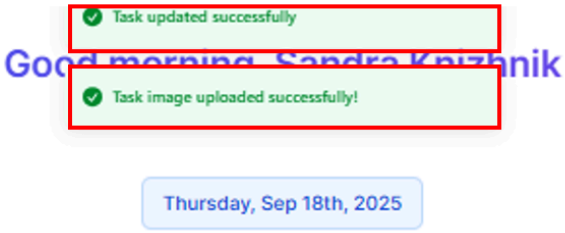
2. Complete Daily Tasks

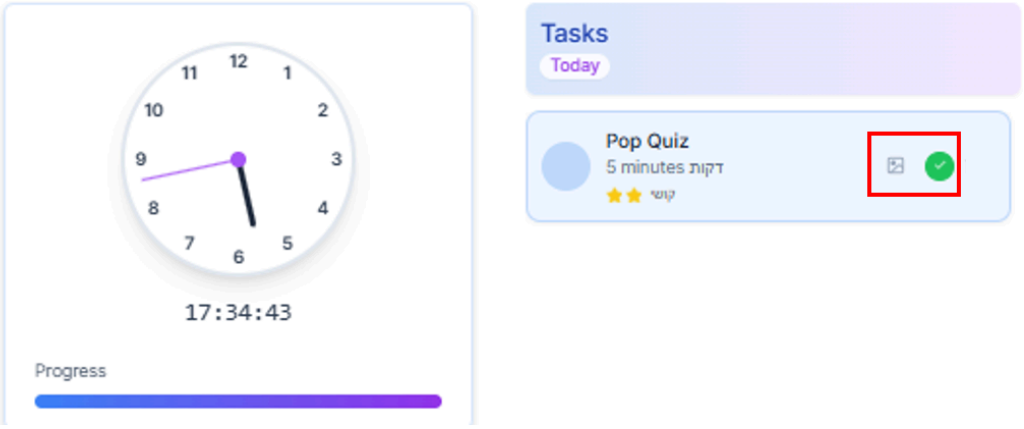
    Perform the academic or behavioral tasks assigned by your teacher.

- For each task, you can activate a **countdown timer** (hourglass icon) adjusted to the expected task duration.

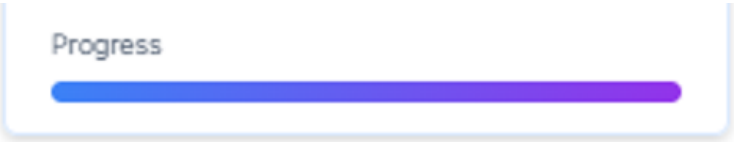


- To mark a task as complete, you must upload a **photo** of the finished work. Only after the upload is confirmed will the task be considered completed.

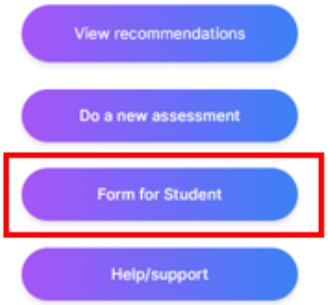




3. Track Progress
   - A **Progress Bar** on the dashboard fills proportionally as tasks are completed. Once all daily tasks are finished, the bar reaches 100%.

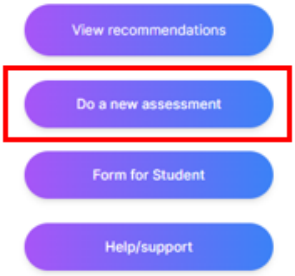


4. Complete Student Questionnaire
    - Students can fill out a self-questionnaire, which contributes to generating personalized recommendations.

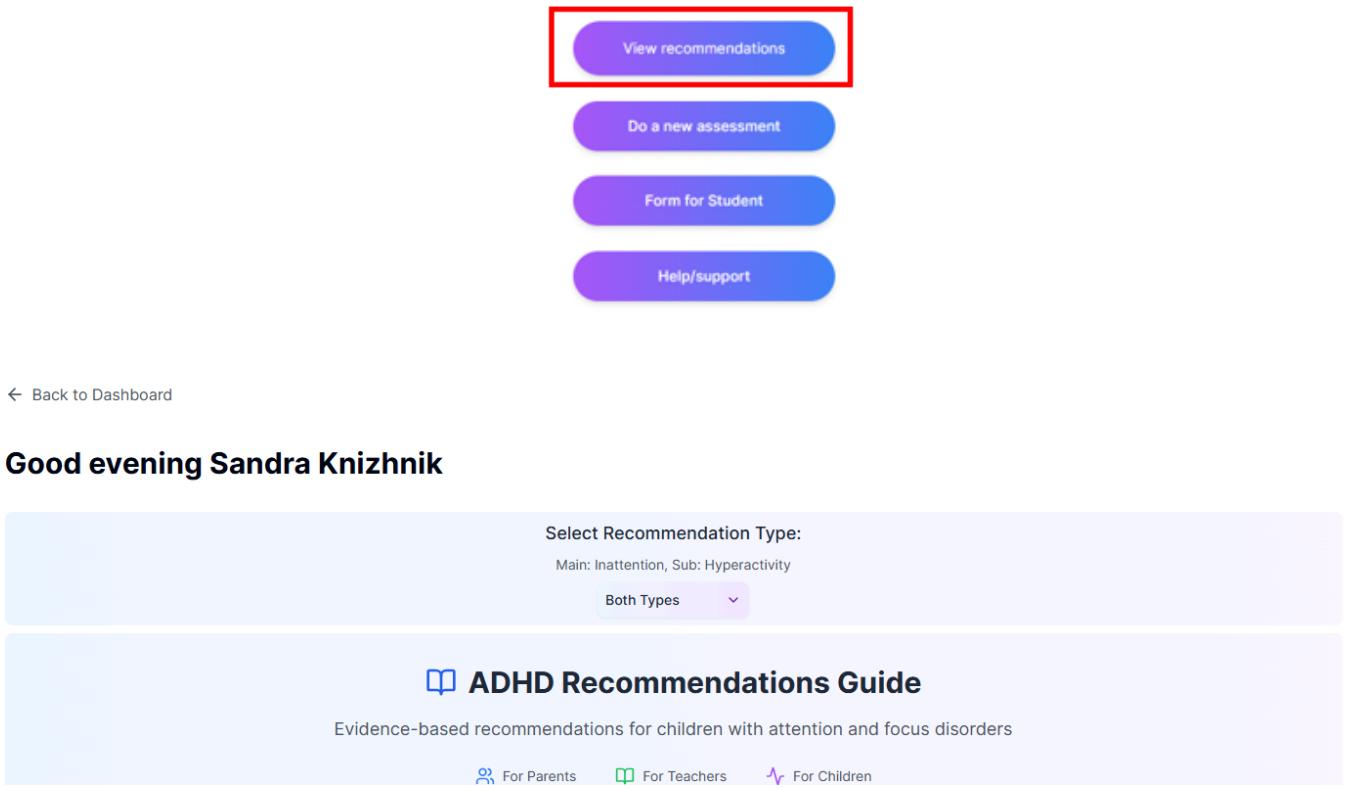


5. Trigger Diagnostic Process
    - If your teacher initiates a diagnostic, a **Do an assessment** button will be enabled in your dashboard.

      By clicking this button, you will be redirected to the external NODUS system to complete the assessment.



6. View Recommendations
    - Students can view recommendations in the same way as teachers and parents, following the instructions in the *Recommendation Display Options* section.



9. Maintenance Guide for BrainBridge

This section includes a full overview of the system structure, environment setup, and procedures for extending and maintaining the BrainBridge platform.

Root Directory

```
brainbridge
├── frontend/          # React + Vite frontend
├── backend/           # Node.js + Express backend
└── README.md
```

Frontend Structure (React + TypeScript).

```
frontend/
├── public/            # Static files (favicon, etc.)
```

```
├── src/
│   ├── assets/          # Images, icons, fonts
│   ├── components/      # Shared UI components
│   │   └── ui/          # Base design system (buttons, cards, toggles)
│   ├── contexts/        # Global React contexts
│   ├── features/        # Feature-based modules
│   │   ├── teacher/     # Teacher dashboards & flows
│   │   ├── parent/      # Parent dashboards & flows
│   │   └── student/     # Student dashboards & flows
│   ├── hooks/           # Custom hooks
│   ├── services/        # API calls to backend
│   ├── App.tsx          # Root component with routes
│   └── main.tsx         # Entry point (Vite)
└── package.json
```

Backend Structure (Node.js + Express) -.

```
backend/
├── src/
│   ├── models/          # Mongoose schemas (User, Task, Recommendation, etc.)
│   ├── controllers/     # Business logic for routes
│   ├── routes/          # Express route definitions
│   ├── middleware/      # Auth & validation
│   ├── pipeline/        # 11-stage recommendation filtering logic
│   ├── utils/           # Helper functions
│   ├── config/          # DB and env configs
```

```
│   ├── app.js            # Express app setup
│   └── server.js         # Entry point
└── package.json
```

9.1 System Environment and Dependencies

Prerequisites

Before running the BrainBridge system, ensure the following software is installed:

- **Node.js** (recommended 16) : Used for backend server and frontend build tools.
- **npm** : To manage dependencies.
- **Python 3.x**: Required for running the Nodus diagnostic service.
- **Git**: To clone the repository and manage updates via version control.

Verify installation:

```
node -v

npm -v

python3 --version

git --version
```

Environment Setup Steps(how to run the project

1. **Clone the repository:**

```
git clone https://github.com/nil-adar/BrainB.git

cd BrainB
```

2. **Install dependencies:**

```
cd BrainB&& npm install
```

3. **Set up .env files for backend:**

```
MONGODB_URI=...
```

4. **Run the backend server:**

Project Folder > npm run server

4.1 Expected output:

```
> vite_react_shadcn_ts@0.0.0 server
> ts-node --project tsconfig.node.json src/server/server.ts

Server running on port 5000
✅ Connected to MongoDB database: BrainB
Connected to MongoDB successfully
```

5. **Run the frontend:**

Project Folder > npm run server

5.1 Expected output:

```
> vite_react_shadcn_ts@0.0.0 dev
> vite

The CJS build of Vite's Node API is deprecated. See h

  VITE v5.4.10  ready in 994 ms

  → Local:   http://localhost:8081/
  → Network: http://192.168.1.102:8081/
  → press h + enter to show help
```

6. **Run the Nodus:**

Project Folder > cd Nodus > py -3.10 manage.py runserver

6.1 Expected output:

dropout (Dropout)	(None, 16, 64)	0
conv1d_1 (Conv1D)	(None, 14, 128)	24,704
batch_normalization_1 (BatchNormalization)	(None, 14, 128)	512
max_pooling1d_1 (MaxPooling1D)	(None, 7, 128)	0
dropout_1 (Dropout)	(None, 7, 128)	0
flatten (Flatten)	(None, 896)	0
dense (Dense)	(None, 128)	114,816
dropout_2 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 64)	8,256
dropout_3 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 4)	260

```
Total params: 149,860 (582.27 KB)
Trainable params: 148,676 (580.77 KB)
Non-trainable params: 384 (1.50 KB)
C:\Users\nilad\AppData\Local\Programs\Python\Python310\lib\site-packages\keras\src\saving\saving_lib.py:576: UserWarning: Skipping variable loading for optimizer
'adam', because it has 2 variables whereas the saved optimizer has 30 variables.
  saveable.load_own_variables(weights_store.get(inner_path))
System check identified some issues:
```

9.2 Installation of Custom Software

Backend Architecture

- Node.js with Express
- MongoDB via Mongoose
- RESTful API structure
- Role-based authentication

Nodus Diagnostic Service

- Python-based external service
- Triggered by the backend to process diagnostics
- Requires separate setup (Python runtime, models)

9.3 Maintenance and Extension Procedures

Routine Maintenance

- **Pull updates from GitHub**:

```
git pull origin main
```

- **Redeploying the System (Railway)**

```
git add .

git commit -m "Your commit message"

git push origin main

After pushing to GitHub, railway will automatically deploy.
```

Extending the System

When adding a new feature new form type, dashboard view, or recommendation logic):

1. **Update MongoDB model (if needed)** - add schema field
2. **Create/extend API route** in /routes/

3. **Write controller logic**
4. **Frontend**: Add component under src/pages or src/components
5. **Connect to the backend using services/ layer**
6. **Test full flow**

Example Development Task: Adding a New Form Type

- Add a new schema in Form.js
- Add a POST route in forms.js
- Update controller formController.js
- Create a new form page in the frontend
- Add form validation + state management using React hooks

Overview

Today BrainBridge runs completely on Railway - no need for a VPS or manual server setup anymore.

The code comes directly from our private GitHub repo.
Every time we push to main, Railway builds and deploys everything automatically.

If we need, we can also click Redeploy manually in the Railway dashboard.
The site uses the custom domain brain-bridge.net.

A CNAME record points the domain to our Railway app.

Railway also gives us a free SSL certificate (so the site is secure with HTTPS).

The backend and frontend run together on the same Express app, so we don't have to deal with CORS or two separate servers

This setup makes it super easy to keep the site updated - we just push to GitHub and Railway takes care of the rest. No more SSH, tmux, or manual restarts.

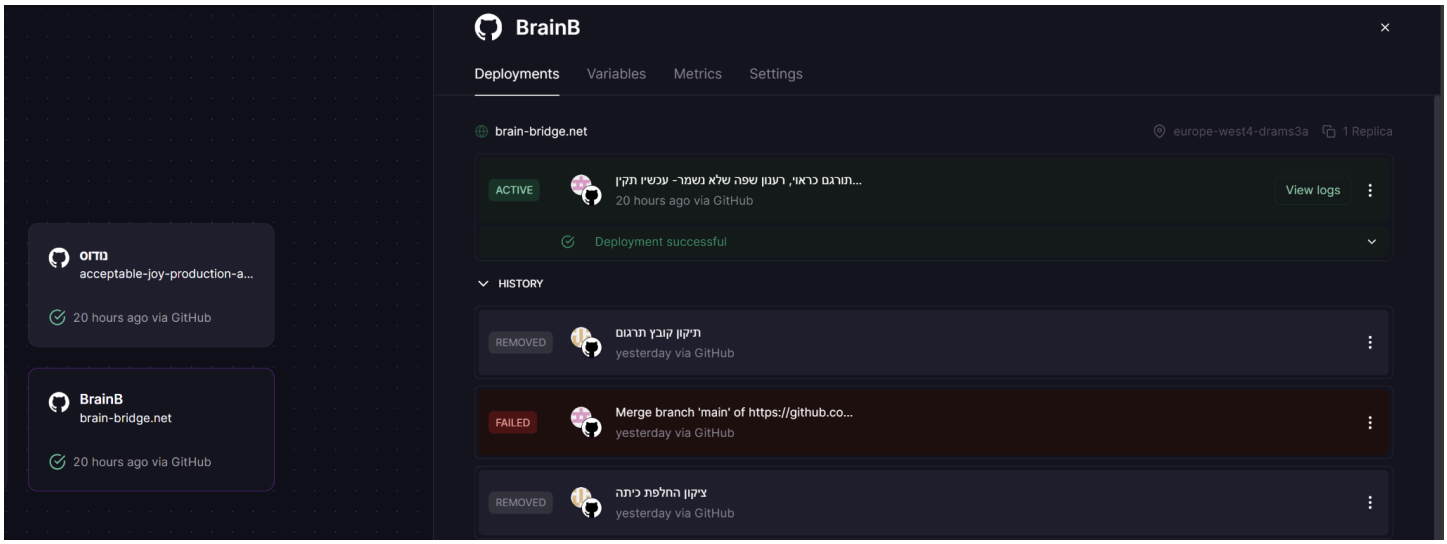
Appendix - Railway Dashboard

The screenshot below shows the **Railway dashboard** where we manage everything for BrainBridge:

- **BrainB Service** - runs the main Express backend and serves the built React frontend.

- **Nodus Service** - a separate Python-based service used for running ADHD diagnostics and sending results back to the backend.
- **Deployments View** - shows current deployment status (green = active) and history of previous builds.
  Logs - allows checking build logs and live application logs for both BrainB and Nodus.
- **Variables** - stores environment variables securely for both services (database URIs, API keys, etc.).
- **Metrics** - shows memory and CPU usage for each service in real-time.

With this setup, both BrainBridge and Nodus are managed from one place - no SSH or manual restarts needed.
All updates are applied automatically when pushing to GitHub, or manually via the **Redeploy** button in Railway.



Managing the App on Railway

We manage BrainBridge completely from the **Railway dashboard**.

- We can see if the app is running and check its status.
- View live logs for both the BrainB service and the Nodus service.
- Trigger a redeploy with one click if we want to deploy manually.
- Edit environment variables (database URLs, API keys) in a secure way.
- Monitor memory and CPU usage to make sure the app is healthy.

Everything is done in the browser - no server logins or manual commands are needed

10. System Testing Guide & Evaluation Credentials

This section provides comprehensive testing scenarios and pre-configured user accounts to facilitate thorough system evaluation. The test cases cover all major workflows across teacher, student, and parent roles, demonstrating BrainBridge's complete functionality.

10.1 Test User Credentials

All test accounts are pre-configured in the development database with different completion states to demonstrate various system workflows.

Role	name	phone	ID	Password	Student(s) Associated	Status
Teacher	ניל אדר	0501234567	312167216	123456	All students	Full access
Parent (with recommendations)	אביעד כהן	0507654321	067487850	123456	נועה כהן, רון כהן	Complete profile
Student	רון כהן	050-2222212	312121054	123456		Full access (Diagnosis: Main-Impulsivity, Sub-Inattention)
Student	נעה כהן	050-1111111	057473324	123456		Full access (Diagnosis: Combined)
Parent (No ADHD + recommendations)	דוד גרוס	0524887291	126533891	123456	אורי גרוס, שמחה גרוס	Partial completion: ☐ parent questionnaire for אורי גרוס
Student	שמחה גרוס	053-4823497	403022261	password 123		Full access (Diagnosis: No ADHD)
Student	אורי גרוס	052-4672497	**ID**: 412055871 **Email**: ori@student.school.com	password 123		Partial completion: ☐ needed Registration ☐ student questionnaire ☐ teacher questionerie ☐ parent questionerie ☐ Nodus assessment

10.2 Testing Scenarios

Test Suite 1: Teacher Workflow

TC-T01: Teacher Login & Dashboard Access

Objective: Verify successful teacher authentication and dashboard loading
Steps:

1. Navigate to BrainBridge login page
2. Enter credentials of : ניל אדר
3. Click "Login"

Expected Result:

- ☐ Successful login redirect to teacher dashboard
- ☐ Dashboard displays class list and daily schedule
- ☐ Greeting message shows teacher name
- ☐ Current date displayed in selected language

TC-T02: Add Personal Event to Schedule

Objective: Verify teacher can create personal calendar events
Steps:

1. From teacher dashboard, click **"Add Personal Event"**
2. Fill in event details:
    - Time Range: 10:00 - 11:00
    - Subject: "Meeting"
    - Topic: "Parent-Teacher Conference"
    - Location: "Room 205"
3. Click **"Add Event"**

Expected Result:

- ☐ Event appears in daily schedule
- ☐ Event persists after page refresh

TC-T03: Add New Student to Class

Objective: Verify teacher can register new students

Steps:

1. Click **"Add New Student"** button
2. Fill in student details:
    - First Name: "Test"
    - Last Name: "Student"
    - ID Number: "123456789"
    - Extra Time: 25% (optional)
3. Click **"Add Student"**

Expected Result:

- ☐ Student card appears in class roster
- ☐ Confirmation message displayed
- ☐ **Verify in MongoDB:** New student document created in users collection

---

TC-T04: Enable NODUS Diagnostic for Student

Objective: Verify teacher can initiate external assessment

Steps:

1. Select student card (e.g., "אורי גרוס")
2. Click **"Create new assessment"**
3. Verify student name auto-populates
4. Click **"Start Assessment"**

Expected Result:

- ☐ Diagnostic session created
- ☐ **Verify in MongoDB:** New document in diagnosticsessions collection with status "pending"
- ☐ Student dashboard now enabled **"Do an assessment"** button

---

TC-T05: Fill Teacher Questionnaire (Complete)

Objective: Verify full questionnaire submission

Steps:

1. From student card , click **"Fill student questionnaire"** (e.g., "אורי גרוס")
2. Complete all 24 questions (select varied responses)
3. Click **"Next"** through all pages
4. Submit questionnaire on final page

Expected Result:

- ☐ Success message: "Questionnaire submitted successfully"
- ☐ Progress indicator (in “view recommendation” for student) shows "Teacher Form: Completed"
- ☐ **Verify in MongoDB:** New document in answers collection with role "teacher"

TC-T06: Partial Questionnaire Abandonment

Objective: Verify system handles incomplete questionnaires

Steps:

1. Start filling teacher questionnaire for another student
2. Answer only 5 questions
3. Close browser tab or navigate away without submitting

Expected Result:

- ☐ A pop-out question “Are you sure?” while asking to abandon
- ☐ No submission recorded
- ☐ **Verify in MongoDB:** No new document created for this session
- ☐ Next login shows questionnaire as "Missing" (in “view recommendation” for student)

TC-T07: Verify Missing Forms Notification

Objective: Confirm dynamic form status updates

Steps:

1. Navigate to **"View recommendations"** for student with incomplete forms
2. Observe the popup modal

Expected Result:

- ☐ Red warning popup appears: "Complete Required Forms"
- ☐ Lists specific missing items (e.g., "Parent Form: Missing")
- ☐ Progress bar shows correct completion percentage (e.g., "2/4 forms completed")

After completing one form:

- ☐ Popup updates to show 3/4 completed
- ☐ Previously "Missing" item now shows "Completed" with green checkmark
- ☐ Returns back to teacher dashboard

TC-T08: Assign Task to Single Student

Objective: Verify individual task assignment

Steps:

1. From student card, click **"Daily Task Update"**
2. Fill task details:
    - Title: "Math Homework"
    - Notes: "Complete exercises 1-10"
    - Date: Today
    - Time: 30 minutes
    - Difficulty: ⭐⭐ (2 stars)
    - Color : blue
3. **Ensure "Assign to all students" toggle is OFF**
4. Select specific student
5. Click **"Add Task"**

Expected Result:

- ☐ Task created successfully
- ☐ **Only selected student** sees the task in their dashboard
- ☐ Other students do NOT see this task

TC-T09: Assign Task to All Students in Class

Objective: Verify bulk task assignment

Steps:

1. Click **"Daily Task Update"**
2. Fill task details:
    - Title: "Reading Assignment"
    - Notes: "Chapter 5"
    - Time: 20 minutes
    - Stars: ⭐⭐⭐ (3 stars)
    - Color : yellow
3. **Toggle "Assign to all students in class" ON**
4. Click **"Add Task"**

Expected Result:

- ☐ Task appears for **all students**
- ☐ **Verify by logging in as multiple students:** All see the same task

---

TC-T10: View Student Recommendations as Teacher

Objective: Verify teacher can access student recommendations

Steps:

1. Select student with completed forms (e.g., "Sandra Knizhnik")
2. Click **"View recommendations"**
3. Observe greeting message and recommendations

Expected Result:

- ☐ Header displays: **"Good [morning/afternoon/evening] - You are viewing as teacher for [Student Name]"**
- ☐ Recommendations load successfully
- ☐ Three category cards displayed: Nutrition, Physical Activity, Environmental Changes

Compare with parent/student view:

- ☐ Same recommendations content
- ☐ Different role-specific greeting
- ☐ PDF export shows identical content

TC-T11: Teacher-Parent Chat Communication

Objective: Verify bidirectional messaging

Steps:

1. From student card, click **"Contact parent"**
2. Type message: "Daniel showed great progress this week. Please review his tasks."
3. Click **"Send Message"**
4. **Switch to parent account**
5. Check notifications (bell icon)
6. Reply to teacher message
7. **Switch back to teacher account**
8. Verify reply appears in conversation

Expected Result:

- ☐ Teacher message sent successfully
- ☐ Parent receives notification
- ☐ Parent can view and reply
- ☐ Teacher sees parent reply
- ☐ Conversation thread preserved
- ☐ **Verify in MongoDB:** Messages saved in messages collection

---

TC-T12: Teacher Logout

Objective: Verify secure logout functionality

Steps:

1. Click profile icon
2. Select **"Logout"**

Expected Result:

- ☐ Redirected to login page
- ☐ Session cleared

Test Suite 2: Student Workflow

TC-S00: Student Resister

Objective: Verify student authentication
Steps:

1. Log in as:אורי גרוס

Expected Result:

- ☐ Successful Registration

TC-S01: Student Login

Objective: Verify student authentication
Steps:

2. Log in as:אורי גרוס

Expected Result:

- ☐ Successful login to student dashboard
- ☐ Greeting: "Good [morning/afternoon], [Student Name]"
- ☐ Dashboard shows assigned tasks(if teacher has assigned), clock, and progress bar

---

TC-S02: Attempt NODUS Diagnostic (Enabled)

Objective: Verify student can access enabled assessment
Precondition: Teacher has created assessment for this student
Steps:

1. Click **"Do a new assessment"** button
2. Observe redirection

Expected Result:

- ☐ Student redirected to external NODUS system
- ☐ NODUS diagnostic loads successfully

- ☐ Student can complete assessment

TC-S03: Attempt NODUS Diagnostic (Not Enabled) for student רון כהן

Objective: Verify appropriate error handling
Precondition: No assessment created for this student
Steps:

1. Try to access diagnostic (button should be disabled and show error)

Expected Result:

- ☐ Button is greyed out
- ☐ Message: "No diagnostic assessment available. Please contact your teacher."

TC-S04: Fill Student Questionnaire

Objective: Verify student self-assessment submission
Steps:

1. Click **"Form for Student"**
2. Complete all 18 questions
3. Submit questionnaire

Expected Result:

- ☐ Success message displayed
- ☐ **Verify in MongoDB:** New document in answers collection with role "student"
- ☐ Form status updates to "Completed" (in “view recommendation” for student)

TC-S05: Send Help Request to Teacher

Objective: Verify student can initiate communication
Steps:

1. Click **"Help/support"** button
2. Type message: "I need help with the math assignment."

3. Click **"Send"**

Expected Result:

- ☐ Message that shows that message was sent successfully
- ☐ Teacher receives notification - not expected since has not been developed.

TC-S06: Complete Task Without Image Upload

Objective: Verify image requirement enforcement

Steps:

1. View assigned task (e.g., "Math Homework")
2. Click task to start countdown
3. Click **green checkmark** to mark as complete WITHOUT uploading image

Expected Result:

- ☐ **Error message:** "Please upload a photo of your completed work."
- ☐ Task remains incomplete
- ☐ Progress bar does NOT update

TC-S07: Complete Task With Image Upload

Objective: Verify successful task completion

Steps:

1. Click camera icon on task card
2. Upload image file (any format)
3. Confirm upload
4. Click green checkmark

Expected Result:

- ☐ Success message: "Task updated successfully"
- ☐ Task marked as complete
- ☐ Progress bar increases
- ☐ **Verify in MongoDB:** Task status updated in tasks collection

---

TC-S08: Student Logout

Objective: Verify logout functionality

Steps:

1. Click profile icon → **"Logout"**

Expected Result:

- ☐ Redirected to login page
- ☐ Session cleared

Test Suite 3: Parent Workflow

TC-P01: Parent Login & Dashboard View

Objective: Verify parent can access their children's profiles

Steps:

1. Log in as: דוד גרוס

Expected Result:

- ☐ Dashboard displays **only** children associated with this parent
- ☐ Each child shown as separate card
- ☐ Buttons available: "Contact Teacher", "Fill questionnaire", "View recommendations"

TC-P02: Initiate Chat with Teacher (Per Child)

Objective: Verify parent-teacher communication per student

Steps:

1. Click **"Contact Teacher"** for "שמחה גרוס"
2. Send a message: "When is the next parent-teacher meeting?"
3. **Repeat for second child** "אורי גרוס"
4. Send different message

Expected Result:

- ☐ Each conversation is **separate and student-specific**
- ☐ Teacher receives **two distinct notifications**
- ☐ Conversations do NOT mix between children

TC-P03: View Child Recommendations

Objective: Verify parent can access child's personalized recommendations (הורה: אביעד כהן)

Steps:

1. Click **"View recommendations for [נעה כהן]"**
2. Observe greeting message

Expected Result:

- ☐ Header: **"Good [time] - You are viewing as parent for [Child Name]"**
- ☐ Recommendations load successfully
- ☐ Content identical to teacher/student views

TC-P04: Fill Parent Questionnaire for Specific Child (הורה: דוד גרוס)

Objective: Verify parent can complete child-specific questionnaire for

Steps:

1. Select child card
2. Click **"Fill questionnaire for [אורי גרוס]"**
3. Complete all 29 questions
4. Submit

Expected Result:

- ☐ Questionnaire submitted successfully
- ☐ **Verify in MongoDB:** Document stored with correct studentId and role "parent"
- ☐ Form status updates for that specific child only

TC-P05: Fill Parent Questionnaire for Specific Child (הורה: דוד גרוס)

Objective: Verify parent can complete child-specific questionnaire for
Steps:

5. Select child card
6. Click **"Fill questionnaire for [אורי גרוס]" - make sure to check the trauma question and some allergies in foods**
7. Complete all 29 questions
8. Submit

Expected Result:

- ☐ Questionnaire submitted successfully
- ☐ **Verify in MongoDB:** Document stored with correct studentId and role "parent"
- ☐ Form status updates for that specific child only
- ☐ Recommendations include professional help
- ☐ Nutrition card recommendations excludes food that were checked as allergic for child

---

Test Suite 4: Cross-Cutting Concerns

TC-CC01: Language Persistence Across Flow

Objective: Verify language selection persists throughout user session
Steps:

1. Log in as any user
2. Change language to English (from Hebrew)
3. Navigate through multiple pages:
    - Dashboard → Questionnaire → Recommendations → Settings
4. Log out and log back in

Expected Result:

- ☐ Language remains English across all pages
- ☐ **After logout:** Language preference persists on next login
- ☐ All UI elements (buttons, labels, messages) display in selected language

---

TC-CC02: Help Button Context Awareness

Objective: Verify help content adapts to current page

Steps:

1. Navigate to **Dashboard page**
2. Click **Help** icon (question mark)
3. Observe displayed content
4. Navigate to **Recommendations page**
5. Click **Help** icon again

Expected Result:

- ☐ Dashboard help: Shows guidance on how to use dashboard features
- ☐ Recommendations help: Explains recommendation categories
- ☐ Help content is **page-specific** and **language-appropriate**

Test Suite 5: Recommendations Display & Multi-Type Handling

TC-R01: First-Time Multiple Diagnosis Type Selection

Objective: Verify system prompts user to choose display mode when student has multiple ADHD subtypes

Precondition: Student has completed all forms and diagnostic shows multiple types (e.g., Main: Hyperactivity, Sub: Inattention)

Steps:

1. Log in as any role (Teacher/Parent/Student)
2. Navigate to "View recommendations" for a student with multiple diagnosis types
3. First visit: Observe automatic popup

Expected Result:

- ☐ Modal appears with title: "Multiple Types Detected"
- ☐ Message displayed: "Your child's main diagnosis is [MainType]%, but also detected some traits of [SubType]%"
- ☐ Two buttons displayed:
  - ☐ "Show All Recommendations" (Both Types)
  - ☐ "Show Only Main Type"
- ☐ Modal cannot be dismissed without making selection

TC-R02: Main Type Only Display

Objective: Verify recommendations display correctly when "Main Type Only" is selected
Steps:

1. From popup modal, click "Show Only Main Type"

Expected Result:

- ☐ Modal closes
- ☐ Recommendations page displays only recommendations for main diagnosis type
- ☐ Category cards show recommendations filtered to main type
- ☐ Dropdown at top of page shows: " [MainType] Only, both types”
- ☐ Verify PDF:
    - ☐ Open PDF export
    - ☐ Confirm PDF contains only main type recommendations

TC-R03: Both Types Display

Objective: Verify recommendations display correctly when "Both Types" is selected
Steps:

1. From popup modal , click "Show All Recommendations"

Expected Result:

- ☐ Modal closes
- ☐ Recommendations page displays combined recommendations from both types
- ☐ Dropdown at top shows: "Both Types"
- ☐ Category cards contain recommendations from both main and subtype
- ☐ Verify PDF:
    - ☐ Open PDF export
    - ☐ Confirm PDF includes recommendations from both types

TC-R04: Switching Between Display Modes

Objective: Verify user can change display preference after initial selection
Steps:

1. After making initial selection (TC-R02 or TC-R03)
2. Locate dropdown menu at top of recommendations page: "Select Recommendation Type"
3. Click dropdown and select different option:
    - If currently "Main Only" → Select "Both Types"
    - If currently "Both Types" → Select "[MainType] Only"
4. Observe page refresh

Expected Result:

- ☐ Dropdown successfully changes selection
- ☐ Recommendations immediately update to reflect new choice
- ☐ Category cards re-render with filtered content
- ☐ PDF export updates: Downloading PDF after change reflects new selection
- ☐ Preference persists: Logging out and back in maintains selected display mode

---

TC-R05: Single Diagnosis Type (No Popup)

Objective: Verify system does NOT show popup when only one diagnosis type exists
Precondition: Student diagnostic shows single type only (e.g., only Hyperactivity OR only Inattention)

Steps:

1. Log in as any role
2. Navigate to recommendations for student with single diagnosis type (e.g "נעה כהן")

Expected Result:

- ☐ No popup appears
- ☐ Recommendations load immediately
- ☐ No Dropdown menu shown
- ☐ All recommendations displayed for that single type

TC-R06: Tab View vs PDF Content Consistency

Objective: Verify recommendations in category tabs match PDF export content exactly
Steps:

1. Navigate to recommendations page
2. Click "Nutrition Recommendations" tab
3. Note all displayed recommendations (titles + descriptions)
4. Repeat for "Physical Activity" tab
5. Repeat for "Environmental Changes" tab
6. Click "Download PDF" button
7. Open downloaded PDF
8. Compare PDF content to tab view

Expected Result:

- ☐ Nutrition section in PDF contains identical recommendations to Nutrition tab
- ☐ Physical Activity section in PDF matches Physical Activity tab exactly
- ☐ Environmental Changes section in PDF matches Environmental tab exactly
- ☐ No recommendations are missing in either format
- ☐ No extra recommendations appear in PDF that weren't in tabs
- ☐ Order may differ but content is identical
- ☐ Recommendation details match:
    - ☐ Title
    - ☐ Difficulty description
    - ☐ Implementation methods
    - ☐ Examples

TC-R07: Category Filtering Consistency Across Formats

Objective: Verify category filtering works identically in both tab view and PDF
Steps:

1. Select "Main Type Only" display mode
2. Note which recommendations appear in each category tab
3. Download PDF
4. Compare category sections in PDF

Expected Result:

- ☐ Each category in PDF contains same filtered set as corresponding tab
- ☐ If tab shows 5 nutrition recommendations, PDF nutrition section shows same 5
- ☐ Categories are properly labeled in PDF matching tab headers

---

TC-R08: Multi-Language Consistency

Objective: Verify recommendations content remains consistent across language switches
Steps:

1. View recommendations in Hebrew
2. Note 3 specific recommendations from different categories
3. Switch language to English
4. Find the same 3 recommendations

Expected Result:

- ☐ Same recommendations appear in both languages (translated, not different content)
- ☐ Category organization remains identical
- ☐ PDF in Hebrew and PDF in English contain same recommendations (in respective languages)

TC-R09: "No ADHD" Result Display

Objective: Verify system shows appropriate message when no ADHD detected
Steps:

1. Log in as any role
2. Navigate to recommendations for student with "No ADHD" diagnostic result (e.g "שמחה גרוס")

Expected Result:

- ☐ Green message box displayed: "Great News! Based on the assessment conducted, no significant signs of ADHD were detected."
- ☐ No category tabs appear
- ☐ No PDF download button available
- ☐ "Back to Dashboard" button functional

10.3 Important Notes for Evaluators

⚠️**Account Registration Limitation:**

- New user registration requires pre-enrollment in the school database

- Users cannot create accounts independently

- For testing purposes, use only the credentials provided above

🔄 **Data Persistence:**

- All test accounts share a common development database

- Changes made during testing will persist across sessions

- To reset a user's state, contact the system administrator

🔒 **Security Note:**

- These credentials are for testing/demonstration purposes only

- Do not use these accounts in production environments

11. Appendix

11.1 Appendix A - SUS Questionnaire & Detailed Results (Final Score: 83.75)

Table1

Question Number	Question	Type	User Score (1–5)	User Score (1–5)	User Score (1–5)	User Score (1–5)	Adjusted Score
1	I would like to use the BrainBridge platform frequently.	Positive	3	3	2	4	3
2	I found the platform unnecessarily complex.	Negative	3	2	3	3	2.75
3	I thought the recommendation interface was easy to use.	Positive	2	4	4	4	3.5
4	I think I would need technical support to use this system.	Negative	3	4	3	4	3.5
5	The different features (student view, parent/teacher inputs, etc.) felt well integrated.	Positive	4	4	4	4	4
6	I noticed inconsistencies or confusing elements in the interface.	Negative	3	1	2	3	2.25
7	I believe most users could learn how to use BrainBridge very quickly.	Positive	3	4	3	3	3.25
8	I found the platform cumbersome or unintuitive.	Negative	4	3	4	4	3.75
9	I felt confident navigating the different features of the system.	Positive	4	3	4	4	3.75
10	I needed to learn a lot before I could effectively use the system.	Negative	3	4	4	4	3.75
		Total Adjusted Score					33.5
		Final SUS Score					83.75

According to SUS benchmarks, a score of 83.75 is considered Excellent usability, indicating that users found the BrainBridge platform intuitive, consistent, and easy to learn

11.2 Appendix B - Data Sources Overview

To support the recommendation engine, BrainBridge relies on two structured datasets:S

Questionnaires Dataset

- Comprises **3 role-specific questionnaires** (Parent, Teacher, Student), with a total of 71 questions:
    - Parent questionnaire: 29 items
    - Teacher questionnaire: 24 items
    - Student questionnaire: 18 items

Recommendations Dataset

- Contains **64 evidence-based recommendations**, organized into three intervention domains:

- Environmental: 24 recommendations
  Physical activity: 22 recommendations
- Nutritional: 18 recommendations

Research Foundations

- Nutritional recommendations incorporate evidence for micronutrient supplementation, dietary modifications, and elimination diets shown to improve ADHD symptoms [41–44].
- Physical activity recommendations are grounded in meta-analytic evidence demonstrating executive function improvements, attention regulation, and behavioral benefits through structured exercise interventions [45–47].
- Environmental recommendations were derived from established behavioral classroom management principles, token economy systems, and evidence-based intervention strategies specifically validated for ADHD students [48–50].

Availability

- Full structured datasets are provided in the GitHub repository:
  - questionnairesDataSet (CSV/Excel format)
  - RecommendationsDataSet (CSV/Excel format)

These datasets form the empirical backbone of BrainBridge, ensuring that all generated recommendations are systematically mapped to validated diagnostic pathways and anchored in peer-reviewed evidence.

12. Research-Based Threshold Justification

Our threshold selection framework addresses a fundamental challenge in translating probabilistic diagnostic outputs from the NODUS system into clinical decision-making. The approach prioritizes diagnostic sensitivity over specificity, reflecting established clinical evidence that ADHD underdiagnosis represents a more significant concern than overdiagnosis when comprehensive assessment protocols are implemented.

Clinical Decision Philosophy:

The NODUS diagnostic system provides probability distributions across ADHD subtypes, requiring systematic threshold decisions for clinical implementation. Our threshold strategy reflects evidence-based clinical reasoning designed to minimize missed diagnoses while maintaining diagnostic rigor through subsequent assessment stages.

Threshold Implementation:

minNoAdhdVal threshold of 70%: This deliberately conservative threshold for "No ADHD" classification ensures that only cases with very high confidence are excluded from further evaluation. Research consistently documents patterns of ADHD underdiagnosis, particularly among adults and females [38][39], supporting an approach that requires strong probabilistic evidence before ruling out ADHD. This threshold prioritizes sensitivity over specificity for exclusion decisions, ensuring that borderline cases receive comprehensive clinical consideration.

minInattVal threshold of 20%: Traditional DSM symptom thresholds have been shown to inadequately identify adults with clinically significant inattentive symptoms [40]. Matte et al. [39] demonstrated that lowering diagnostic thresholds increased valid ADHD identification by 27% in adults, with newly identified cases showing equivalent functional impairment to traditionally diagnosed cases. This lower threshold addresses documented underdiagnosis patterns, particularly for inattentive presentations that may manifest with lower symptom visibility while maintaining clinical significance.

minHyperVal threshold of 25%: Research indicates that hyperactive-impulsive symptoms require adjusted thresholds for adult populations, as traditional six-symptom requirements exclude clinically significant presentations [40]. This threshold acknowledges that while hyperactive symptoms may be more observable than inattentive symptoms, they also decrease developmentally while retaining clinical relevance [39]. The slightly higher threshold reflects this greater symptom visibility while maintaining clinical sensitivity.

Evidence-Based Rationale:

Our threshold framework embodies clinical principles supported by ADHD assessment literature. Research demonstrates that conventional clinical elevation thresholds typically fall at the 93rd or 98th percentile [41], validating percentile-informed approaches to threshold determination. The multi-stage assessment architecture of our system provides additional safeguards against false positive classifications through questionnaire integration and clinical validation stages.

Implementation Justification:

This threshold strategy prioritizes sensitivity over specificity based on clinical evidence that ADHD underdiagnosis represents a greater public health concern than overdiagnosis when proper assessment protocols are followed [38][39][40]. The approach ensures that potentially affected individuals receive appropriate clinical evaluation rather than premature exclusion, while subsequent assessment stages provide systematic controls for diagnostic accuracy.

Methodological Transparency:

These thresholds represent evidence-informed clinical adaptations rather than empirically derived cutoff points. The specific numerical values reflect systematic integration of published research findings with the practical requirements of probabilistic diagnostic implementation. Future validation studies examining these thresholds against clinical outcomes and functional impairment measures will be essential for refining their clinical utility.

13. Known Limitations

1. **Issue Type:** Bug
   Category: PDF Export

   Description: When generating personalized PDF recommendation files using `@react-pdf/renderer`, Hebrew text occasionally displays corrupted or unreadable characters. The issue does not occur with English content.

   Impact: Partially reduces the readability of Hebrew PDF reports, which may require manual correction or limit usability for Hebrew-speaking users.

   Root Cause: This behavior is linked to incomplete support for right-to-left (RTL) text rendering and Hebrew font encoding within `@react-pdf/renderer`.

   Future Direction:

   - Evaluate custom font embedding with full Unicode/CID support.
   - Investigate community workarounds for RTL handling in `@react-pdf/renderer`.
   - Consider fallback or hybrid solutions (e.g., generating Hebrew PDFs with an alternative rendering library).

References

[1] *The Effects of Social Media on Dopamine Levels. Neurolaunch. Retrieved from* https://neurolaunch.com/social-media-dopamine/

[2] *Smith, J., & Brown, K. (2021). The Effects of Digital Overstimulation on Attention and Learning in Children. Frontiers in Psychology.* https://www.frontiersin.org/articles/10.3389/fpsyg.2021.611155/full

[3] *Viola, A., Balsamo, L., Neglia, J. P., Brouwers, P., Ma, X., & Kadan-Lottick, N. S. (2017). The Behavior Rating Inventory of Executive Function (BRIEF) to Identify Pediatric Acute Lymphoblastic Leukemia (ALL) Survivors At Risk for Neurocognitive Impairment. Journal of pediatric hematology/oncology, 39(3), 174–178.* https://doi.org/10.1097/MPH.0000000000000761

[4] *Biederman, J., DiSalvo, M., Vaudreuil, C., Wozniak, J., Uchida, M., Woodworth, K. Y., ... & Faraone, S. V. (2021). The child behavior checklist can aid in characterizing suspected comorbid psychopathology in clinically referred youth with ADHD. Journal of psychiatric research, 138, 477-484.*

[5] *Reynolds, C. R., & Kamphaus, R. W. (2004). Behavior Assessment System for Children (2nd ed.). AGS Publishing.*

[6] *Moore, D. A., Russell, A. E., Arnell, S., & Ford, T. J. (2017). Educators' experiences of managing students with ADHD: a qualitative study. Child: care, health and development, 43(4), 489-498.*

[7] *Martinez, R., & Wighting, M. (2023). Teacher-Student Relationships: Impact of Positive Behavioral Interventions and Supports. Athens Journal of Education, 10(3), 397-410.*

[8] *ADDitude Editors. (2023). A Brief History of ADHD: From 1798 to Today. ADDitude Magazine. Retrieved from https://www.additudemag.com/history-of-adhd/*

[9] *Government of Israel. (n.d.). Children with ADHD. Retrieved February 8, 2025, from https://www.gov.il/he/pages/children-adhd*

[10] *Psychologim. (n.d.). הפרעות קשב וריכוז. Retrieved February 8, 2025, from https://psychologim.com/הפרעות-קשב-וריכוז/*

[11] *Clalit Health Services. (n.d.). הפרעת קשב וריכוז (ADHD) אצל ילדים: המדריך המלא. Retrieved February 8, 2025, from https://www.clalit.co.il/he/your_health/kids/childs/Pages/tova.aspx*

[12] *Shen, F., & Zhou, H. (2024). Advances in the etiology and neuroimaging of children with attention deficit hyperactivity disorder. Frontiers in Pediatrics, 12, 1400468.*

[13] *Ra, C. K., Cho, J., Stone, M. D., De La Cerda, J., Goldenson, N. I., Moroney, E., ... & Leventhal, A. M. (2018). Association of digital media use with subsequent symptoms of attention-deficit/hyperactivity disorder among adolescents. JAMA, 320(3), 255-263.*

[14] *Lawrenson, J. G., Hull, C. C., & Downie, L. E. (2017). The effect of blue-light blocking spectacle lenses on visual performance, macular health and the sleep-wake cycle: a systematic review of the literature. Ophthalmic and Physiological Optics, 37(6), 644-654.*

[15] *Kostić, J. O., & Randelović, K. R. (2022). Digital distractions: Learning in multitasking environment. Psychological Applications and Trends, 301-304.*

[16] *Saccaro, L. F., Schilliger, Z., Perroud, N., & Piguet, C. (2021). Inflammation, anxiety, and stress in attention-deficit/hyperactivity disorder. Biomedicines, 9(10), 1313.*

[17] *Sycamore, B. Viewing ADHD's Impact on Children's Social Development Through the Lens of Peer Rejection. Family Perspectives, 6(1), 1.*

[18] *Relational Psych Group. (2024). Exploring the Impact of School Stress on Teens' Emotional Well-being. Retrieved from https://www.relationalpsych.group/articles/exploring-the-impact-of-school-stress-on-teens-emotional-well-being*

[19] *Kwon, S. J., Kim, Y., & Kwak, Y. (2018). Difficulties faced by university students with self-reported symptoms of attention-deficit hyperactivity disorder: a qualitative study. Child and adolescent psychiatry and mental health, 12, 1-8.*

[20] *Faraone, S. V., Biederman, J., Spencer, T., Wilens, T., Seidman, L. J., Mick, E., & Doyle, A. E. (2000). Attention-deficit/hyperactivity disorder in adults: an overview. Biological psychiatry, 48(1), 9-20.*

[21] *NeuroLaunch. (n.d.). Is ADHD more common in males or females? Retrieved from https://neurolaunch.com/is-adhd-more-common-in-males-or-females/*

[22] *ADHD Weekly. (2017, September 14). School Suspension Risk Higher for Students with ADHD. Retrieved from https://chadd.org/adhd-weekly/school-suspension-risk-higher-for-students-with-adhd/*

[23] *The Summit Psychology Clinic. (n.d.). ADHD in Men vs. Women: Why Gender Matters in Diagnosis and Treatment. Retrieved from https://www.thesummitpsychology.com/blog/adhd-in-men-vs-women-why-gender-matters-in-diagnosis-and-treatment*

[24] *NeuroLaunch. (n.d.). ADHD in Women vs. Men. Retrieved from* https://neurolaunch.com/adhd-in-women-vs-men/

[25] *Mechler, K., Banaschewski, T., Hohmann, S., & Häge, A. (2022). Evidence-based pharmacological treatment options for ADHD in children and adolescents. Pharmacology & therapeutics, 230, 107940.*

[26] *Clalit. (n.d.). Ritalin (Methylphenidate Hydrochloride) for ADHD Treatment. Retrieved from* https://www.clalit.co.il/he/medical/pharmacy/Pages/ritalin_methylphenidate_hydrochloride.aspx

[27] *Nazarova, V. A., Sokolov, A. V., Chubarev, V. N., Tarasov, V. V., & Schiöth, H. B. (2022). Treatment of ADHD: Drugs, psychological therapies, devices, complementary and alternative methods as well as the trends in clinical trials. Frontiers in Pharmacology, 13, 1066988. doi:10.3389/fphar.2022.1066988*

[28] *Lange, K. W., Lange, K. M., Nakamura, Y., & Reissmann, A. (2023). Nutrition in the management of ADHD: A review of recent research. Current Nutrition Reports, 12(3), 383-394. doi:10.1007/s13668-023-00472-9*

[29] *על הקשר בין תזונה והפרעת קשב - קווים ומחשבות. (n.d.). Retrieved from* https://www.kavim-machshavot.com/adhd-diet

[30] *Den Heijer, A. E., Groen, Y., Tucha, L., Fuermaier, A. B., Koerts, J., Lange, K. W., ... & Tucha, O. (2017). Sweat it out? The effects of physical exercise on cognition and behavior in children and adults with ADHD: a systematic literature review. Journal of Neural Transmission, 124, 3-26. doi:10.1007/s00702-016-1593-7*

[31] *Kankanhalli, A., Taher, M., Cavusoglu, H., & Kim, S. H. (2012). Gamification: A new paradigm for online user engagement.*

[32] *Hamari, J., Koivisto, J., & Sarsa, H. (2014, January). Does gamification work?--a literature review of empirical studies on gamification. In 2014 47th Hawaii international conference on system sciences (pp. 3025-3034). IEEE.*

[33] *Seesaw. (n.d.). Seesaw helps teachers track student progress and organize tasks, supporting students with ADHD. Retrieved from* https://apps.apple.com/us/app/seesaw/id930565184

[34] *Chasanidou, D., Gasparini, A. A., & Lee, E. (2015). Design thinking methods and tools for innovation. In Design, User Experience, and Usability: Design Discourse: 4th International Conference, DUXU 2015, Held as Part of HCI International 2015, Los Angeles, CA, USA, August 2–7, 2015, Proceedings, Part I (pp. 12-23). Springer International Publishing.*

[35] *Liu, C., & Kaeser, P. S. (2019). Mechanisms and regulation of dopamine release. Current opinion in neurobiology, 57, 46-53.*

[36] *Basu, S., & Dasgupta, P. S. (2000). Dopamine, a neurotransmitter, influences the immune system. Journal of neuroimmunology, 102(2), 113-124.*

[37] *Gaz, L. (2025, June). Expert interview on classroom attention challenges and environmental context [Personal communication].*

[38] *Rosenthal, E. A., Broos, H. C., Timpano, K. R., & Johnson, S. L. (2024). Does Emotion-Related Impulsivity Relate to Specific ADHD Symptom Dimensions, and Do the Effects Generalize Across Comorbid Internalizing and Externalizing Syndromes?. Journal of attention disorders, 28(2), 178–188. https://doi.org/10.1177/10870547231210283*

[39] *Sadr-Salek, S., Costa, A. P., & Steffgen, G. (2023). Psychological Treatments for Hyperactivity and Impulsivity in Children with ADHD: A Narrative Review. Children (Basel, Switzerland), 10(10), 1613. https://doi.org/10.3390/children10101613*

[40] *Gomez, R., Stavropoulos, V., Watson, S., Brown, T., & Chen, W. (2022). Inter-relationships between ADHD, ODD and impulsivity dimensions in emerging adults revealed by network analysis: extending the 'trait impulsivity hypothesis'. Heliyon, 8(10), e10712. https://doi.org/10.1016/j.heliyon.2022.e10712*

[41] *Lange, K. W., Lange, K. M., Nakamura, Y., & Reissmann, A. (2023). Nutrition in the Management of ADHD: A Review of Recent Research. Current nutrition reports, 12(3), 383–394. https://doi.org/10.1007/s13668-023-00487-8*

[42] *Pinto, S., Correia-de-Sá, T., Sampaio-Maia, B., Vasconcelos, C., Moreira, P., & Ferreira-Gomes, J. (2022). Eating Patterns and Dietary Interventions in ADHD: A Narrative Review. Nutrients, 14(20), 4332. https://doi.org/10.3390/nu14204332*

[43] *Razavinia, F., Ebrahimiyan, A., Faal Siahkal, S., Ghazinezhad, N., & Abedi, P. (2024). Vitamins B9 and B12 in children with attention deficit hyperactivity disorder (ADHD). International journal for vitamin and nutrition research. Internationale Zeitschrift fur Vitamin- und Ernahrungsforschung. Journal international de vitaminologie et de nutrition, 94(5-6), 476–484. https://doi.org/10.1024/0300-9831/a000809*

[44] *Landaas, E. T., Aarsland, T. I., Ulvik, A., Halmøy, A., Ueland, P. M., & Haavik, J. (2016). Vitamin levels in adults with ADHD. BJPsych open, 2(6), 377–384. https://doi.org/10.1192/bjpo.bp.116.003491*

[45] *Song, Y., Fan, B., Wang, C., & Yu, H. (2023). Meta-analysis of the effects of physical activity on executive function in children and adolescents with attention deficit hyperactivity disorder. PloS one, 18(8), e0289732.* https://doi.org/10.1371/journal.pone.0289732

[46] *Wang, P., Rao, F., Xu, Z., Xing, K., Gao, Y., & Li, D. (2025). Effects of aerobic exercise on executive function in children and adolescents with attention deficit hyperactivity disorder: a systematic review and meta-analysis of randomized controlled trials. BMC sports science, medicine & rehabilitation, 17(1), 257.* https://doi.org/10.1186/s13102-025-01304-1

[47] *Li, D., Li, L., Zang, W., Wang, D., Miao, C., Li, C., Zhou, L., & Yan, J. (2023). Effect of physical activity on attention in school-age children with ADHD: a systematic review and meta-analysis of randomized controlled trials. Frontiers in physiology, 14, 1189443.* https://doi.org/10.3389/fphys.2023.1189443

[48] *Pfiffner, L. J., & Haack, L. M. (2014). Behavior management for school-aged children with ADHD. Child and adolescent psychiatric clinics of North America, 23(4), 731–746.* https://doi.org/10.1016/j.chc.2014.05.014

[49] *Reiber, C., & McLaughlin, T. F. (2004). Classroom interventions: Methods to improve academic performance and classroom behavior for students with attention-deficit/hyperactivity disorder. International Journal of Special Education, 19(1), 1-13.* https://files.eric.ed.gov/fulltext/EJ852038.pdf

[50] *Eiraldi, R. B., Mautone, J. A., & Power, T. J. (2012). Strategies for implementing evidence-based psychosocial interventions for children with attention-deficit/hyperactivity disorder. Child and adolescent psychiatric clinics of North America, 21(1), 145–x.* https://doi.org/10.1016/j.chc.2011.08.012