TECHNISCHE UNIVERSITÄT
CHEMNITZ

Fakultät für Informatik

CSR-25-05

Development of a Retrieval Model based Backend of a Tutoring Agent

Ambu Babu · Ummay Ubaida Shegupta · Wolfram Hardt

August 2025

Chemnitzer Informatik-Berichte

TECHNISCHE UNIVERSITÄT
CHEMNITZ

Time Management Tool Development to Support Self-regulated Learning

Master Thesis

Submitted in Fulfilment of the
Requirements for the Academic Degree
Masters of Engineering

Dept. of Computer Science
Chair of Computer Engineering

Submitted by: Tripti Kumari Shukla
Student ID: 564733
Date: 28.11.2024

Supervising tutor: Prof. Dr. Dr. h. c. Wolfram Hardt
Supervising tutor: Prof. Ummay Ubaida Shegupta

Abstract

This thesis explores the design and development of a time management tool that integrates principles of Self-Regulated Learning (SRL) to support students in online and hybrid learning environments. The tool aims to address critical needs in planning, monitoring, and reflecting on academic tasks, empowering learners to achieve autonomy and adaptability in managing their responsibilities. By combining task scheduling, milestone tracking, and reflective feedback into a unified platform, the tool provides a structured yet flexible framework for enhancing learning outcomes.

Built on the foundation of SRL, the tool incorporates the three key phases of the learning process: planning, execution, and reflection. In the planning phase, it enables students to set achievable goals, create schedules, and prioritize tasks. During execution, it offers real-time progress tracking and milestone monitoring to keep learners aligned with their objectives. In the reflection phase, the tool facilitates self-assessment and adaptive strategy refinement, fostering essential SRL skills such as goal-setting, self-monitoring, and critical thinking.

The proposed solution addresses challenges found in existing fragmented time management systems, which often result in inefficiencies and reduced learner motivation. By integrating core functionalities into a cohesive platform, the tool bridges these gaps, emphasizing usability and adaptability for seamless integration into daily routines. Key features include task scheduling, milestone tracking, reflective feedback, and workload visualization, all designed to enhance strategic planning and reflective practices.

Leveraging modern technologies such as React, Flask, and SQLite, and following a user-centered design approach, the tool ensures an intuitive interface and practical functionality based on feedback from students and educators. Expected outcomes include improved time management, greater learner autonomy, and enhanced academic performance. This innovative approach has the potential to redefine how students plan, manage, and reflect on their learning journeys, contributing significantly to their overall success.

Keywords: Self-Regulated Learning (SRL),Time Management Tool, Learning Management System (LMS), Reflective Learning Practices, Learning Analytics and Visualization

Contents

Contents **6**

List of Figures **9**

List of Tables **12**

List of Abbreviations **14**

1 Introduction **16**
1.1 Self Regulated Learning 18
1.2 Background and Challenges 20
1.3 Motivation 22
1.4 Scope 24

2 Related Work **25**
2.1 Recent Work 25
2.1.1 State-of-the-Technique 25
2.1.2 State-of-the-Art 27
2.1.3 Aligning Focus Area 28
2.1.4 Need Gap 29
2.2 Open Research Question 32
2.3 Approach 34

3 Methodology **35**
3.1 Concept 35
3.2 User Flow Diagram 38
3.3 Design Prototype 41
3.4 Implementation Plan 44
3.4.1 Design Phase 45
3.4.2 Development 46
3.4.3 Testing 46
3.4.4 Deployment and Integration 47

4 Implementation **48**
4.1 Situation and Problem 48
4.2 Interactive Time Management Web Tool 49
4.2.1 Architecture Overview 49

4.2.2 Designing the Database Schema . . . 52
4.2.3 Development and Implementation . . . 54
4.2.4 Integration of a Time Management Tool within the ARC Tutoring Workbench . . . 58
4.2.5 Testing and Deployment: . . . 58
4.3 Survey Structure and Format . . . 60
4.3.1 Questionnaire . . . 60
4.3.2 Survey Format Description . . . 63
4.3.3 Scoring . . . 65

5 Results and Evaluation . . . 67
5.1 Results . . . 67
5.1.1 Output of The Time Management Webtool . . . 67
5.1.2 Survey and Responses . . . 71
5.2 Evaluation and Analysis . . . 76
5.3 Revisiting Research Questions and Discussions . . . 80
5.3.1 Research Goal 1: Self-Regulated Learning (SRL) . . . 80
5.3.2 Research Goal 2: Usability and time-tracking . . . 84
5.3.3 Research Goal 3: User Satisfaction . . . 89
5.3.4 Research Goal 4: Seamless Integration and Broad Applicability 91

6 Conclusion . . . 95
6.1 Conclusions . . . 95
6.2 Future Work . . . 98

Bibliography . . . 100

List of Figures

1.1 Phases of Self Regulated Learning 18
1.2 Common issues faced by students in their learning journey 20
1.3 Motivation to build a Comprehensive Self Regulated Learning Tool Design 22

2.1 Recent work has been distributed in different domains 25
2.2 A comparision of limitations of State-of-the-Technique 29

3.1 A Usage Flow Diagram for a Learner Perspective, defining the flow of action and scenarios on the Web App 38
3.2 Prototype for Time Planner for visualization Purposes via MS Excel 41
3.3 Stages of Implementation Phases 44
3.4 Detail plan of tasks distribution in different Implementation Stages . 45

4.1 Architecture ARC Tutoring Workbench integrated with Time Planner Webtool Component 49
4.2 UML Diagram of the Database Schema of Task Table for Time Management Webtool component 52
4.3 UML Diagram of the Database Schema of ARC Tutoring Workbench 54
4.4 Code Snippet to implement database schema for the time management tool via Models.py 55
4.5 Code Snippet of an example of an endpoint for adding tasks: 56
4.6 Code Snippet of database Intialization 57
4.7 Code Snippet of API Integration using Blueprint 58
4.8 Code Snippet on how we tested various functionalities of the code while output generation 59
4.9 Depiction of Graphical Output and Analysis of after Survey Responses and Scoring and 66

5.1 Full page view of final output of Time Management Webtool 67
5.2 Output focused to display the time tracking features 68
5.3 Output focused to display of Add task and Add Milestones functions 68
5.4 Output focused to display of Edit Task and Delete Task functions . . 69
5.5 Snippet focused to display the Table SQLite3 Database file via Database Browser 70
5.6 Mean and Standard Deviation for USE Items (USE 1.1–1.6) 77
5.7 Mean and Standard Deviation for SRL Items (SRL 1.1–1.3) 78

5.8 Median and Mode for Independent Items (CKU, UF, FL, OS) 79
5.9 Frequency Distribution for Independent Items 80
5.10 Heatmap of SRL and USE Response 81
5.11 Trend Analysis: SRL and USE Response over time 82
5.12 Correlation Matrix Heatmap 85
5.13 Scatter Plot with Regression 87
5.14 Net Promoter Score (NPS) Analysis: Assessing User Satisfaction . . . 90
5.15 Stacked Bar Chart: Gender-Based Breakdown of Responses 92
5.16 Radar Chart: Key Features of Usability and Integration 93

List of Tables

2.1 Focus Areas in Learning Enhancement Tools 27

5.1 Gender and Age Group Codes 71
5.2 Nationality Codes 72
5.3 Frequency and 5-Point Scale Codes 72

List of Abbreviations

SRL Self-Regulated Learning

LMS Learning Management System

SQL Structured Query Language

API Application Programming Interface

CRUD Create, Read, Update, Delete

UML Unified Modeling Language

SUS System Usability Scale

IoT Internet of Things

AI Artificial Intelligence

MS Microsoft

JDK Java Development Kit

VM Virtual Machine

TP Time Planner

USE1.3 Usability Scale Element 1.3

USE1.6 Usability Scale Element 1.6

SRL1.1 Self-Regulated Learning Metric 1.1

SRL1.3 Self-Regulated Learning Metric 1.3

ARC Academic Resource Center

GLASS Global Learning Analytics Support System

SNAPP Social Network Analysis and Pedagogical Practice

CKU Cognitive Knowledge Unit

UF User Feedback

FL Feature List

OS Operational System

M2LADS Multimodal Learning Analytics Dashboard System

SAM Student Activity Monitor

NPS Net Promoter Score

1 Introduction

The thesis, "Time Management Tool Development to Support Self-Regulated Learning", aims to create a digital tool that helps students manage their time effectively and enhance their ability to learn independently. Self-regulated learning (SRL) emphasizes planning, monitoring, and reflecting on one's learning process. As education shifts towards online and hybrid models, students must take greater responsibility for their academic tasks, yet many struggle to balance learning with personal and professional commitments.

This thesis focuses on developing an integrated time management tool that aligns with the principles of SRL. Unlike existing tools that function in isolation, such as calendar apps or task trackers, this tool will provide a holistic solution by combining task scheduling, milestone tracking, progress visualization, and reflective features. By doing so, it aims to address inefficiencies caused by fragmented tools and foster autonomy, goal-setting, and strategic planning.

Designed with adaptability and usability in mind, the tool will support diverse student needs, offering customizable task lists, milestone management, and visualization options like weekly and daily views. It will empower students to take control of their educational journey while supporting both short-term goals and long-term planning. Ultimately, this thesis seeks to bridge the gap between theoretical SRL frameworks and practical learning needs, enhancing students' academic success.

Managing time effectively is crucial for academic success, especially in the context of self-regulated learning (SRL). With the growing shift toward online and hybrid education models, students are now required to take more responsibility for their learning. This involves planning study schedules, setting realistic goals, keeping track of progress, and reflecting on their results. However, transitioning to this self-directed style of learning can be challenging. Many students find it difficult to balance their academic work with personal and professional responsibilities.

While there are numerous digital tools available to help students, these tools often work independently, addressing only specific needs. For example, a calendar app might help with scheduling, while a separate tool could be used for collecting feedback. This lack of integration can create inefficiencies, reduce motivation, and make it harder for students to gain a complete picture of their learning progress.

To tackle these challenges, this thesis focuses on creating a time management tool designed specifically around the principles of self-regulated learning. The goal is to provide students with an all-in-one solution that not only helps them manage their time effectively but also supports them in taking full control of their educational journey.

In the next section, lets deep dive into understanding what do we mean by self regulated learning (SRL)in detail.

1.1 Self Regulated Learning

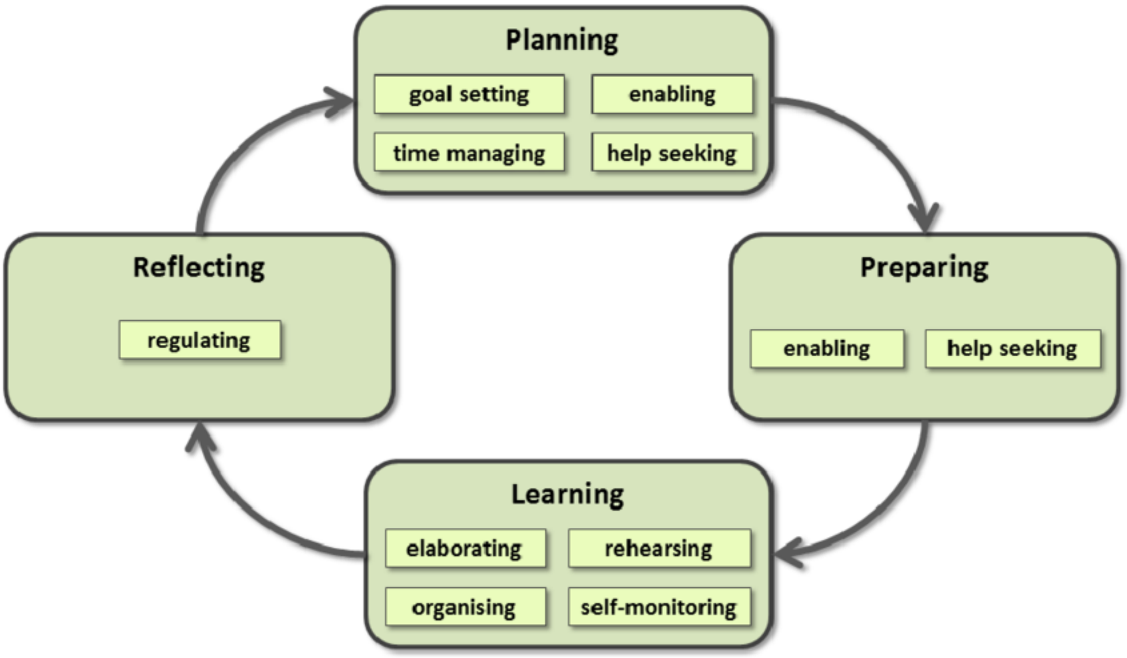

Figure 1.1: Phases of Self Regulated Learning

Self-regulated learning (SRL) is a process by which learners take active control of their educational experiences through planning, monitoring, and evaluating their progress. Rooted in educational psychology, SRL emphasizes autonomy, self-discipline, and strategic thinking.

It is a cyclical process as depicted figure below involving three key phases: planning, where learners set goals and develop strategies; execution, where tasks are performed while monitoring progress; and reflection, where outcomes are evaluated to inform future actions.

SRL provides a framework for addressing the fragmented nature of traditional learning tools. By integrating cognitive and metacognitive activities, such as goal setting, time management, and critical reflection, SRL enables students to develop a deeper understanding of their learning processes. This methodology not only enhances academic performance but also fosters skills like resilience, adaptability, and independent thinking, which are essential in today's dynamic world.

The role of technology in SRL is increasingly significant. Digital tools have the potential to bridge gaps in existing educational practices by offering features that support the entire SRL cycle. For instance, an effective tool can enable students to set realistic goals, track time spent on tasks, and reflect on progress in a systematic manner. Moreover, such tools can facilitate minimal yet meaningful interventions from instructors, allowing them to guide students without micromanaging their efforts.

In this thesis, the development of a time management tool is envisioned as a means to operationalize the principles of SRL. By providing a cohesive, integrated platform, the tool aims to empower learners to take charge of their educational journeys, fostering both immediate and long-term success.

1.2 Background and Challenges

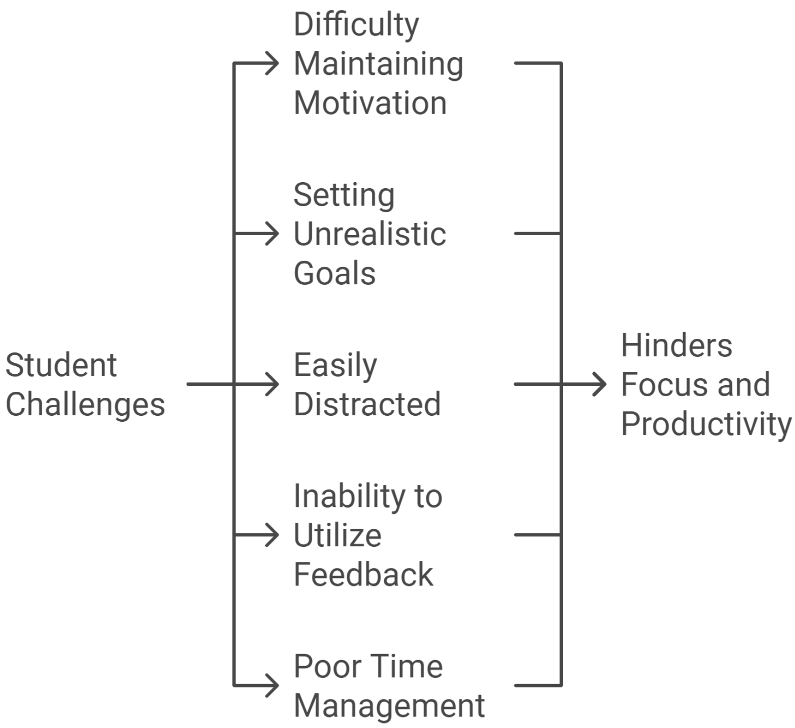

Figure 1.2: Common issues faced by students in their learning journey

The learning journey of students is often marked by various difficulties, as outlined in recent studies and practical observations. A key challenge is maintaining motivation and persistence over time. Many learners struggle to stay engaged, particularly when faced with overwhelming workloads or when lacking a clear sense of direction. Another common issue is the tendency to set unrealistic goals. While ambitious targets can be motivating, they often lead to frustration and demotivation if they are not attainable within the given time frame.

Distractions from social media, entertainment, and other non-academic activities further hinder students' ability to focus. The digital age has made it easier than ever for learners to lose precious time on unproductive activities, detracting from their ability to meet academic objectives. Additionally, many students face difficulties in effectively utilizing feedback to refine their learning strategies. Without a structured approach to incorporating feedback, learners often miss out on opportunities to improve their performance and outcomes.[1]

Time management is another significant hurdle. Poor time allocation, procrastination, and inefficient use of time are recurring themes in students' struggles. Together, these challenges highlight the need for a holistic solution that not only addresses individual issues but also integrates them into a unified framework. The aim is to create a tool that not only helps students overcome these barriers but also fosters habits and skills that support lifelong learning.

1.3 Motivation

In today's fast-paced learning environments, students are increasingly required to take control of their learning processes. This shift toward self-regulated learning (SRL) demands a structured approach where learners can independently plan, organize, execute, and evaluate their progress. However, the tools available to support such endeavors are often fragmented, leaving learners to juggle multiple platforms to manage their educational journey. [1]This lack of integration creates challenges in capturing the full spectrum of SRL, where cognitive, metacognitive, and motivational aspects must work in harmony.

Figure 1.3: Motivation to build a Comprehensive Self Regulated Learning Tool Design

For instance, a learner might use one tool for scheduling tasks, another for tracking deadlines, and yet another for recording feedback or reflecting on progress. This fragmented experience not only increases cognitive load but also prevents a holistic understanding of their learning process. Incomplete or disjointed data makes it harder for students to identify patterns, refine strategies, and stay motivated over time.

The motivation for this thesis stems from the need to address these challenges by designing a comprehensive time management tool that integrates seamlessly into the SRL framework. Such a system would consolidate data across various phases of learning, offering a unified platform to support planning, goal setting, time tracking, and reflective practices. By incorporating meta-cognitive elements such as critical reflection and progress monitoring the tool can enable students to gain deeper insights into their learning behaviors. This, in turn, empowers them to make informed decisions, adapt their strategies, and achieve greater autonomy in their educational pursuits.

Additionally, an effective SRL-supporting system benefits not only learners but also educators. With features like real-time progress monitoring and minimal intervention requirements, professors can support students more efficiently, focusing on guidance rather than micromanagement. [1]Ultimately, such a tool aims to bridge the gap between fragmented digital systems and the holistic needs of self-regulated learners, paving the way for more effective, efficient and engaging learning experiences.

1.4 Scope

Designing, creating and assessing a time management application that promotes self-regulated learning (SRL) is the main topic of this thesis. With features like work scheduling, milestone monitoring, and progress visualization, the program seeks to help students manage their academic assignments efficiently. It is intended to support individual students in online and hybrid learning environments by assisting them with learning planning, tracking, and reflection.[1] The tool's usefulness and conformity to SRL principles will be assessed in the thesis, offering a workable way to improve students' independence and strategic planning. To ensure a clear and targeted approach to accomplishing the research objectives, collaborative tools and non-academic task management are outside the purview of this study.

2 Related Work

2.1 Recent Work

In recent years, numerous systems and applications have been developed to monitor and report students' time management and learning activities. These tools vary in their target users—learners, educators, or both—and in their approaches to data presentation, ranging from raw data displays to advanced predictive analytics. The referenced sources collectively highlight the technological advancements, methodologies, and predictive algorithms that have been explored in the field of monitoring and reporting systems for learning platforms. Below is a compilation of 15 notable studies and systems from 2020 to 2024 that contribute to this field:

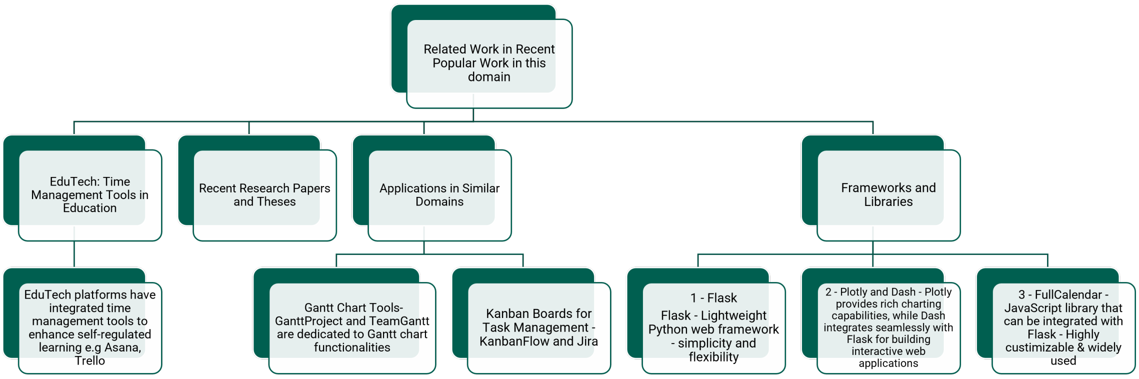

Figure 2.1: Recent work has been distributed in different domains

2.1.1 State-of-the-Technique

Technologies Explored
The field of learning analytics and monitoring systems has utilized a wide range of technologies to track, analyze, and visualize learner activities.

Learning Management Systems (LMS): Platforms like Moodle, Blackboard, and custom dashboards (e.g., Student Success System, GLASS, SAM) have been instrumental in logging learner interactions, resource usage, and time allocation. These systems help facilitate tracking and reporting, often combining raw data with analytics to enhance monitoring capabilities [2, 3] [2], [3], [4].

Visualization Tools: Visualization tools play a critical role in representing learner data meaningfully. Systems such as StepUp! and SNAPP use graphs and charts to show insights into learners' progress, social interactions, and time spent on tasks. These visualizations promote reflection and awareness among learners and instructors [7], [8], [11].

Multimodal Data Integration: Platforms like M2LADS integrate data from multiple sources to provide a holistic view of learner activities. These systems combine cognitive and metacognitive dimensions, such as goal setting, planning, and task monitoring, to support comprehensive learning analytics [9], [10].

AI and IoT: Recent advancements, such as AI-driven analytics and Internet of Things (IoT) technologies, have been employed to enhance real-time monitoring.[1] These technologies facilitate personalized feedback and adaptive learning experiences to improve student engagement and outcomes [4].

Mobile and Web Applications: Systems like Course Signal and LearnTracker utilize mobile-friendly interfaces, enabling learners to track activities and receive real-time notifications or alerts about their performance thresholds [7], [10], [13].

Learner Activities Explored Monitoring systems capture diverse activities, reflecting cognitive and metacognitive processes:

Time Management: Tools like SAM and GLASS have focused extensively on tracking time spent on activities or resources. These systems provide insights into how learners allocate their time and identify engagement patterns [2], [4], [8].

Task and Resource Usage: Systems monitor learners' usage of specific resources (e.g., videos, documents), frequency of interactions, and completion rates for tasks or milestones. For instance, StepUp! tracks both task completion and collaborative efforts, helping students and instructors understand participation levels [7], [10].
Behavioral Data: Tools such as SNAPP focus on social interactions in forums or group discussions, analyzing participation trends and their correlation with learning outcomes [7].
Progress and Performance: Dashboards like Learning Analytics Dashboard and Course Signal monitor academic progress through assessments and resource usage. These systems provide comparative data for learners, allowing them to evaluate their performance relative to their peers [2], [3], [8].

Offline Activities: Emerging systems emphasize capturing offline learning behaviors, such as preparation and reflective practices. Including offline activities ensures that the learning process is comprehensively monitored across various stages [11], [12].

Focus Area	Description	Limitations	Citations
Time Management	Tracking and analyzing time allocation to identify engagement patterns, helping learners manage their schedules effectively.	Fails to connect time usage with overarching learning goals, focusing more on surface-level metrics.	[2], [4], [8]
Task and Resource Usage	Monitoring resource usage and task completion rates, providing insights into productivity and learning behaviors.	Focuses on resource tracking without directly linking to learning outcomes or deeper analytics.	[7], [10]
Mobile and Web Applications	Real-time notifications and performance tracking through tools like Course Signal, ensuring timely interventions.	Limited focus on offline activities or reflective processes, narrowing the scope of tracking.	[7], [10], [13]
Visualization Tools	Graphical tools like StepUp! and SNAPP represent progress, interactions, and engagement visually, enhancing reflection and awareness.	Static visualizations lacking interactivity, limiting learners' ability to explore data for deeper insights.	[7], [8], [11]

Table 2.1: Focus Areas in Learning Enhancement Tools

2.1.2 State-of-the-Art

Now in the subsection below of our thesis, we will see the highlights the importance of combining cognitive and metacognitive monitoring with predictive analytics to create a more adaptive and effective learning environment. The referenced works collectively illustrate the ongoing efforts to enhance SRL through innovative methodologies.

Predictive Algorithms and Future Performance

Several tools and systems have implemented predictive analytics to estimate future outcomes and identify at-risk learners:

Ensemble Models and Predictive Analytics: The Student Success System employs ensemble predictive models to identify students at risk of failure. By analyzing historical data and engagement patterns, these models suggest actionable interventions [3], [4], [8].

Data Mining Techniques: Many tools, such as GLASS and LearnTracker, incorporate data mining to identify patterns in learner behavior. These techniques predict future activities and provide insights for optimizing learning strategies [2], [4].

Machine Learning Algorithms: AI-powered platforms like n-Gage leverage machine learning models to predict emotional, behavioral, and cognitive engagement. These systems provide real-time feedback, enhancing learner performance and motivation [9], [10].

Comparative Analytics: Systems like GLASS allow learners to compare their progress with their peers, identifying gaps and opportunities for improvement [2], [4], [7].

Threshold-Based Alerts: Tools like Course Signal notify learners and instructors when performance metrics fall below a predefined threshold. These alerts serve as early interventions to improve engagement and outcomes [7], [10].

The integration of technologies such as AI, multimodal dashboards, and data mining algorithms has significantly advanced monitoring and reporting systems in education. Predictive models have proven effective for identifying at-risk students and offering tailored interventions. Visualization tools and mobile platforms have enhanced accessibility and usability, promoting self-regulation and metacognition in learning. However, challenges remain in fully integrating diverse data sources and addressing offline activities, which are crucial for a holistic Self-Regulated Learning (SRL) experience [4], [9], [13].

2.1.3 Aligning Focus Area

Focus on Learning enhancement Tool regards to Time Management,Task and Resource Usage and Visualization Tool

To align with the focus of our thesis, we emphasize the need to enhance learning tools specifically in the domains of Time Management, Task and Resource Usage, Mobile and Web Applications, and Visualization Tools. These areas are pivotal in addressing the challenges of supporting learners' Self-Regulated Learning (SRL) processes.

Title of our thesis is **"Time Management Tool Development to Support Self-regulated Learning"** and hence our study aims to improve Time Management by providing tools that not only track time allocation but also align it with overarching learning goals. In the realm of Task and Resource Usage, we aim to move beyond surface-level monitoring to deliver actionable insights that connect resource use with learning outcomes. Mobile and Web Applications will play a crucial role in real-time tracking and notifications, ensuring learners receive timely interventions. Finally, by enhancing Visualization Tools, we seek to develop interactive and dynamic visual representations that foster deeper reflection and awareness, empowering learners to make informed decisions in their SRL journey.

This thesis will address the limitations in current systems and propose integrated, learner-centric solutions to fill these critical gaps.

2.1.4 Need Gap

There is a notable need gap in the existing research and tools for time management systems supporting learners' Self-Regulated Learning (SRL) journeys. While significant progress has been made in developing monitoring and reporting systems, several limitations and challenges persist that highlight the need for more effective, holistic, and integrated tools tailored to learners' SRL needs. Below, We have elaborated on these gaps and their implications:

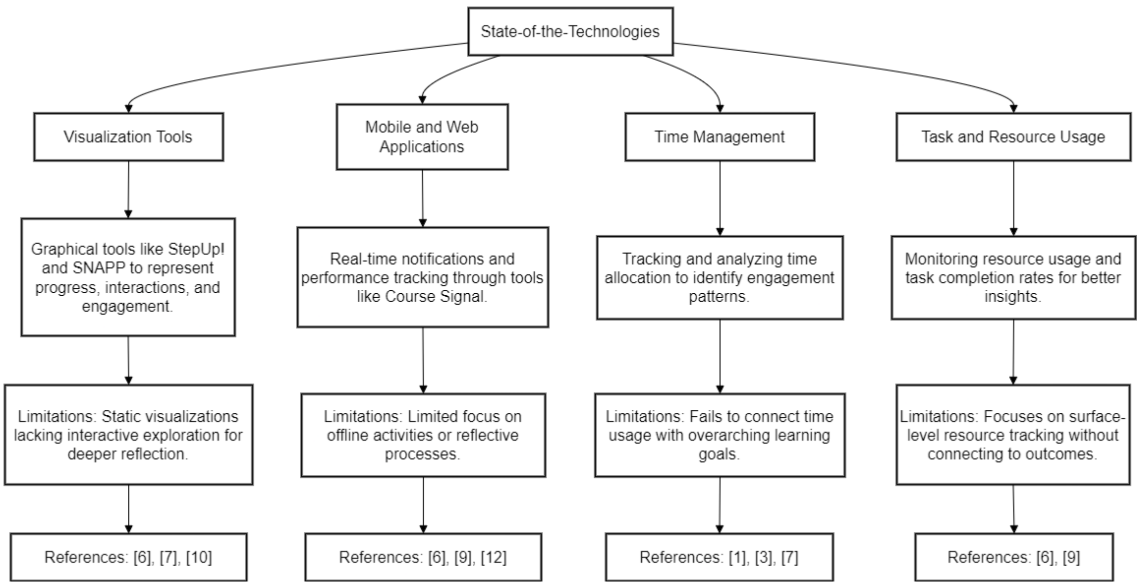

Figure 2.2: A comparision of limitations of State-of-the-Technique

Comparision:

1. Limited Scope of Existing Tools

Many of the existing tools and systems, such as Moodle dashboards, GLASS, and SNAPP, focus on specific aspects of the learning process, such as tracking login/logout times, time spent on tasks, or interactions in forums [2], [4], [8]. While these metrics are useful, they only cover the surface-level cognitive activities of learners. SRL, however, requires deeper integration of metacognitive processes, such as goal setting, planning, monitoring, and reflection, which are often overlooked in current tools [7], [9], [13].

Why This Matters:

Time management in SRL is not just about tracking time spent on tasks but also understanding how learners allocate their time strategically to meet their learning goals. Tools need to support learners in aligning their time management behaviors with their personal goals, which requires capturing and analyzing both cognitive and metacognitive processes.

2. Insufficient Emphasis on Visualization for Reflection

Visualization is a powerful tool for supporting reflection and awareness. While systems like StepUp! and GLASS include graphs and charts, these visualizations are often limited to static metrics, such as time spent or scores [7], [8]. Dynamic and interactive visualizations that allow learners to explore their time management patterns in depth are still underexplored.

Why This Matters:

Effective visualization can transform abstract data into actionable insights. Tools should enable learners to interact with their data—zooming into specific time periods, comparing planned versus actual schedules, and identifying trends over time—to foster deeper reflection and better decision-making.

3. Limited Learner-Centric Features

Most tools, such as LOCO-Analyst and Student Success System, are designed primarily for teachers to monitor learners' performance and identify at-risk students [4], [6], [8]. While some tools, like StepUp! and GLASS, provide visualizations for learners, these are often simplistic and fail to empower learners to take control of their SRL journey fully [7], [8].

Why This Matters:

Learners are the primary agents in SRL. Tools need to provide intuitive, learner-friendly features, such as personalized dashboards, time-allocation recommendations, and reflective prompts. A learner-centric approach is essential to foster self-awareness, self-monitoring, and self-reflection.

4. Lack of Real-Time Feedback and Interaction
Current tools often rely on static reports or post-hoc analytics, which do not provide learners with the immediate feedback they need to make real-time adjustments. For example, systems like SAM and Course Signal offer alerts or reports based on predefined thresholds but do not adapt dynamically to learners' evolving needs [7], [13].

Why This Matters:

Real-time feedback is critical for effective time management in SRL. A tool that can notify learners of potential time mismanagement, suggest immediate adjustments, or provide motivational nudges can enhance their ability to self-regulate and stay on track.

Based on the identified gaps, there is a clear need for a **holistic time management tool** that:

- Integrates all phases of SRL, from planning to reflection.
- Combines online and offline learning activities for comprehensive tracking.
- Provides real-time feedback and adaptive recommendations based on predictive analytics.
- Empowers learners with personalized, learner-centric features and dynamic visualizations.
- Bridges cognitive and metacognitive dimensions to support goal setting, monitoring, and reflective practices.

This tool would go beyond simply monitoring activities to actively supporting learners in developing and refining their SRL strategies, addressing the limitations of existing systems [2]-[16]. The development of this next-generation tool represents a critical step toward enhancing the SRL experience for learners in diverse educational contexts.

2.2 Open Research Question

Research Goal 1:

"How can a true self-regulated learning autonomous learning tool can be achieved ?"

Self-regulated learning (SRL), in which students take charge of their own learning processes by establishing goals, tracking their progress, and commenting on their results, has been the focus of educational innovations in recent years. However, a digital platform alone is not enough to create a true autonomous self-regulated learning tool ; an environment that actively promotes and leads students to become independent learners is also necessary. Although efforts have been made to develop tools that promote self-regulation, few have been able to strike a balance between direction and independence, which makes it difficult to maintain motivation and engagement. This begs the question: How can a learning tool that is really autonomous and self-regulated be created to promote long-term, independent learning without sacrificing the support that education needs?

Research Goal 2:

"Which features enabling students to track time and ensuring high usability and robust time tracking?"

Features for User-Friendliness and Efficient Time Monitoring in Student Learning Resources Effective time management is essential for academic performance, particularly in situations involving self-directed study. Educational technologies with efficient time-tracking capabilities can help students meet deadlines, create realistic calendars, and improve their time management abilities. However, adding these functions without sacrificing the tool's usefulness is still difficult. Problems with usability may make students less inclined to use time-tracking features, which might compromise the tool's instructional value. Which particular elements can help students efficiently manage their time while maintaining a high degree of usability, robustness, and engagement within time-tracking functionalities? This raises an essential question.

Research Goal 3:

"How can a tool be designed for user satisfaction, emphasizing usability and ease of use ?"

Usability and perceived ease of use are frequently directly related to user satisfaction in educational technologies. Students are more likely to stick with a tool that feels natural to them, is easy to use, and effectively helps them achieve their objectives in a learning setting. But creating a tool that is both highly usable and user-satisfied can be difficult, particularly when it comes to tools that are meant to accommodate a variety of learning styles and preferences. Developing successful educational technologies requires an understanding of the connection between usability, user pleasure, and the entire learning experience. This raises the challenge of how to prioritise usability and ease of use in a learning tool's design while maintaining crucial instructional features in order to maximise user happiness.

Research Goal 4:

"What design ensure seamless integration and broad applicability?"

The necessity of tools that can adapt to different learning environments and easily interact with current systems has been brought to light by the recent growth of educational technology. Learning tools must not only offer useful features but also be readily compatible with a variety of educational frameworks and technology in order to be widely usable and accepted by institutions. For developers looking to produce solutions that are adaptable enough for various educational contexts while preserving functional coherence and simplicity of integration, this demand presents a substantial barrier. Thus, a crucial research question is raised: Which design techniques can guarantee the smooth integration and wide application of educational tools across various institutions, learning contexts, and technology ecosystems?

2.3 Approach

To address these research questions, our methodology will involve a multi-phase approach combining user-centered design, iterative prototyping, and empirical evaluation. Initially, we will conduct a comprehensive literature review to understand existing frameworks for self-regulated learning, usability in educational tools, and effective time-tracking mechanisms. Building on these insights, we will engage with target users—students and educators—through interviews and surveys to gather requirements and expectations. Based on these findings, we will develop an initial prototype focusing on core features that promote autonomy, usability, and seamless integration. This prototype will undergo iterative testing and refinement, with feedback loops to ensure it aligns with user needs and enhances the learning experience. Finally, we will evaluate the tool's effectiveness through controlled usability studies, analyzing metrics related to user satisfaction, engagement, and learning outcomes to validate its impact and guide further improvements. Through this structured approach, we aim to achieve a comprehensive solution that addresses the goals of autonomy, usability, and applicability in educational settings.

3 Methodology

3.1 Concept

In this section, we will demonstrate how the time management tool applies the principles of planning, execution, and reflection to help students manage their academic tasks. We will use real-life examples to show how the tool assists in managing short-term deadlines, handling complex projects, and fostering self-regulated learning. Through these examples, we aim to showcase how the tool empowers students to take control of their academic work and improve their strategies over time.

Let's imagine Student A, an undergraduate engineering student facing a busy semester. Student A is enrolled in several courses and uses the time management tool to stay organized, track progress, and develop better learning habits.

For managing managing Overlapping Tasks, we can understand it through examples where Student A has three urgent tasks to complete in one week: a Physics lab report (Task X) due in three days, a Computer Science coding project (Task Y) due in five days, and preparation for a Mathematics midterm exam (Task Z) happening in a week. To handle this workload, Student A logs into the time management tool and inputs all three tasks with their respective deadlines.
The tool automatically organizes the tasks on a calendar and prioritizes them based on their urgency. It suggests that Student A focus on the Physics lab report (Task X) first, followed by the coding project (Task Y). The Mathematics midterm (Task Z) is flagged as "Medium Priority," allowing Student A to allocate some study time for it throughout the week.

Each evening, the tool sends reminders to ensure that Student A is on track with Task X. After completing the lab report, Student A logs it as finished in the tool. At this point, the tool asks reflective questions such as, "Did you finish this task on time?" and "Were you able to manage distractions effectively?" Through this reflection, Student A realizes that starting earlier could have reduced last-minute stress. This insight helps them plan better for future tasks.

For managing Long-Term Milestones, we can understand it through examplere where Student A is also working on a capstone project, which is a major academic milestone. The project involves several phases: submitting a proposal, collecting data, performing analysis, and writing the final report. Using the time management tool, Student A creates a detailed timeline for these phases. For example, the proposal submission is due by March 1 (Milestone 1), data collection by April 10 (Milestone 2), analysis by May 1 (Milestone 3), and the final report by June 1 (Milestone 4). The tool displays these milestones as a Gantt chart, showing how they depend on each other—for instance, analysis cannot begin until data collection is complete.

Midway through the project, a piece of equipment malfunctions, delaying data collection. Student A updates the timeline in the tool, which automatically adjusts the deadlines for subsequent milestones. Despite this setback, the recalibrated plan keeps Student A on track. When the data collection phase is completed, the tool prompts a reflection: "What challenges did you face during this phase?" and "How could these challenges be avoided in the future?" Student A documents the lessons learned, preparing them better for the analysis phase.

For implementing a true Self-Regulated Learning in an Online Course, we can understand it through examples where in addition to assignments and milestones, Student A is enrolled in an online certification course on Advanced Machine Learning. This six-week course requires disciplined self-study. Student A uses the time management tool to set a goal: "Achieve a grade above 90 percentage." The tool breaks this goal into smaller tasks, such as "Watch Week 1 Videos," "Complete Quiz 1," and "Review Practice Problems." Each task is assigned a deadline and added to the weekly calendar.

As Student A progresses through the course, the tool tracks how much time is spent on each activity. For example, after Week 1, the tool generates a report: "You spent 5 hours watching videos but only 1 hour on quizzes. Consider increasing your quiz practice to reinforce learning." This feedback helps Student A adjust their study plan for Week 2.

At the end of the week, the tool prompts reflective questions: "Did you achieve your weekly goal?" and "What adjustments will you make for next week?" By answering these questions, Student A becomes more aware of their strengths and areas for improvement. Over time, this iterative process of setting goals, monitoring progress, and reflecting helps Student A develop the autonomy and discipline needed for independent learning. By the end of the course, Student A not only achieves the desired grade but also gains confidence in managing self-paced learning.

Through these examples, it is clear that the time management tool effectively integrates the principles of planning, execution, and reflection. Whether managing short-term deadlines, navigating complex milestones, or fostering self-regulated learning, the tool serves as a reliable companion for Student A. By helping students organize their tasks, adjust to challenges, and reflect on their progress, the tool not only improves academic performance but also nurtures essential skills for lifelong learning.

3.2 User Flow Diagram

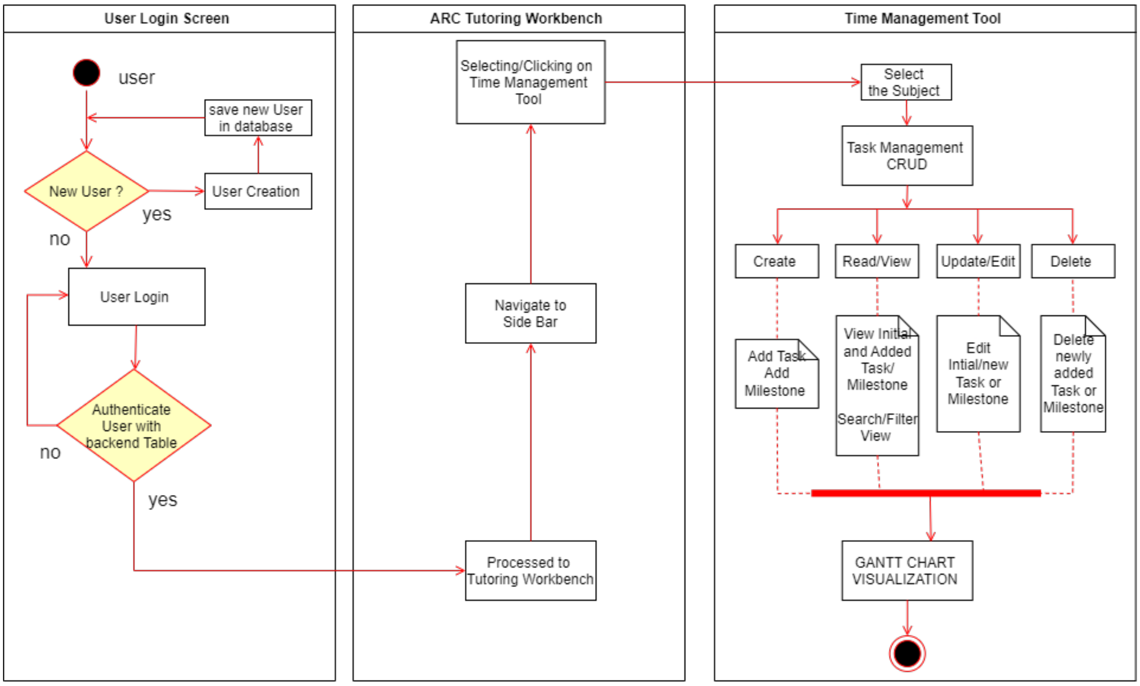

Figure 3.1: A Usage Flow Diagram for a Learner Perspective, defining the flow of action and scenarios on the Web App

This user flow diagram provides a comprehensive overview of a student's interaction with the time management tool within a tutoring workbench system. Through a structured pathway of user authentication, task management, and progress visualization, this system empowers students to engage in systematic academic planning and milestone tracking. The integration of CRUD functionality with Gantt chart visualization promotes a high level of task management autonomy, aligning with engineering principles of usability, efficiency, and adaptability.

The following user flow diagram outlines the interaction process for a student utilizing a time management tool integrated into a tutoring workbench system. This flow encompasses user authentication, task management, and the visualization of progress using a Gantt chart. Each component of this flow is essential for enhancing the student's ability to manage tasks and monitor academic progress within a structured tutoring framework.

1. User Login Screen

The entry point of this system is the User Login Screen, which facilitates user authentication and registration. The flow begins by querying whether the user is a new or returning individual.

New User Registration: If the user is identified as new, a pathway initiates the creation of a user profile, which is subsequently stored within a backend database. This action enables future logins and access continuity for the user.
Returning User Login: In cases where the user has an existing profile, the login sequence advances directly to the User Authentication phase. This phase verifies credentials against the backend database to confirm identity and authorize access to the tutoring workbench.
Upon successful authentication, the user is granted access to the Tutoring Workbench Dashboard.

2. ARC Tutoring Workbench Navigation

Within the ARC Tutoring Workbench Dashboard, the user has access to various academic support tools. A key functionality is the Time Management Tool, accessible via a side navigation panel. Selecting this tool allows the student to manage tasks and track milestones related to their academic progress.

3. Time Management Tool - Task Management and CRUD Operations

Upon entry into the Time Management Tool, users gain access to a Task Management Interface that supports complete CRUD (Create, Read, Update, Delete) functionality. This interface is designed to enable students to perform a range of task-related operations that contribute to effective time and resource management within their academic framework

Create: This function allows students to add new tasks and milestones. The creation of tasks involves specifying parameters such as task name, description, start and end dates, and associated milestones. This functionality is essential for planning upcoming academic activities and setting interim goals.

Read/View: The system provides a read functionality that enables students to review both newly added and previously established tasks. This feature may also support filtering or searching tasks, allowing students to efficiently locate and examine specific milestones and deadlines.

Update/Edit: The update functionality allows students to modify task parameters, such as changing the start or end date, updating milestones, or adjusting the task's status based on progress. This dynamic modification is essential for adjusting plans in response to changing academic priorities or progress.

Delete: The delete function empowers students to remove tasks or milestones that are no longer relevant, thereby keeping the task interface uncluttered and focused on current academic priorities. The delete operation may include a confirmation step to prevent accidental task removal.

Each operation within the Task Management module directly impacts the visualization of the student's progress.

4. Gantt Chart Visualization

Upon the completion of any CRUD operation, the system updates a Gantt Chart Visualization. This Gantt chart provides a graphical timeline that displays each task and milestone relative to the overall academic schedule. The visualization enables students to see their progress at a glance, identify upcoming deadlines, and monitor the completion of individual tasks and milestones.

This Gantt chart serves as an integral part of the time management tool, providing a visual feedback loop that enhances the student's ability to plan, monitor, and adjust their academic activities in real-time.

3.3 Design Prototype

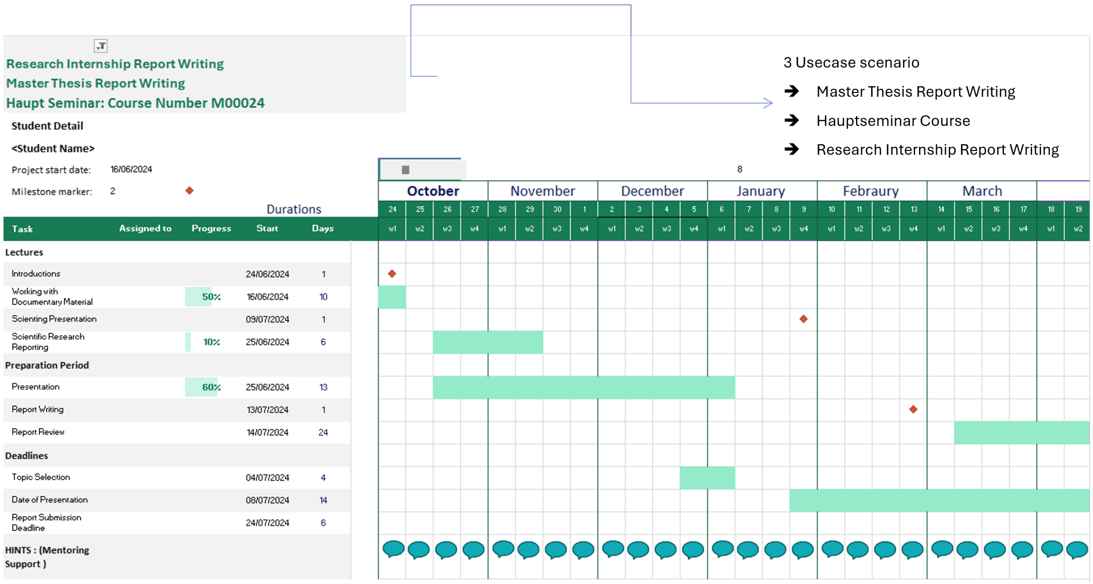

Figure 3.2: Prototype for Time Planner for visualization Purposes via MS Excel

The planned Time Management Planner for students is designed to comprehensively address the specific needs of learners in managing their academic tasks and milestones while fostering a structured and visually intuitive approach. The following are the envisioned features and functionalities, described in a detailed manner to align with the requirements of a scientific academic paper.

Main Features and Functionalities
The Time Management Planner aims to provide an organized and user-friendly interface where students can manage their academic tasks effectively. Upon selecting a subject, the planner will populate a default set of pre-defined tasks tailored to the selected subject's curriculum. These default tasks serve as the foundation of the planner, ensuring that students are guided through the standard requirements of their coursework.

To maintain consistency and structure, students will not be permitted to delete these default tasks. However, they will have the flexibility to personalize their schedules by editing key attributes, such as the start date, end date, progress percentage, and task status. This level of customization allows students to adapt the planner to their unique pacing and progress without compromising the essential structure of their subject-specific tasks.

Custom Task and Milestone Management In addition to default tasks, students will have the ability to add their own tasks beyond those pre-defined in the planner. These custom tasks can reflect individual preferences or additional academic responsibilities not covered in the default set. Students will have full control over these custom entries, with the ability to edit or delete them as needed.

The planner also supports the creation and management of milestones, which serve as significant markers in a student's learning journey. Milestones can be added independently or in association with tasks. To visually distinguish milestones from standard tasks, they will be represented as flags on the planner rather than progress bars. This unique representation provides a clear and intuitive way for students to identify critical achievements or deadlines at a glance. Like custom tasks, milestones can be edited or deleted, offering students dynamic control over their academic roadmap.

Visualization and Timeframe

To enhance usability and support diverse planning needs, the planner will include an option for students to toggle between week-view and day-view modes. The week-view mode offers a broader perspective, helping students visualize their workload over an extended period, while the day-view mode provides a focused and detailed breakdown of daily responsibilities. This dual-view functionality ensures that students can plan effectively, whether for long-term goals or immediate priorities.

Furthermore, the planner will feature a default timeline spanning eight months from the designated start date. This extended timeframe is particularly beneficial for students engaged in semester-based or year-long academic programs, allowing them to plan, track, and reflect on their progress throughout the duration of their studies.

The proposed features align with the principles of Self-Regulated Learning (SRL) by fostering autonomy, adaptability, and strategic planning. Providing pre-defined tasks ensures that students have a structured starting point while enabling them to personalize their schedules to suit individual needs. The ability to add and manage custom tasks and milestones supports metacognitive processes, such as goal-setting and reflection, which are critical components of SRL.

The visualization options, such as the week and day views, cater to different planning styles and enhance cognitive engagement by offering clarity in workload distribution. The eight-month timeline accommodates realistic academic cycles, ensuring that the tool remains relevant and practical for extended periods of use.
By incorporating these functionalities, the Time Management Planner will serve as an effective learning enhancement tool, empowering students to take control of their

academic responsibilities while supporting their self-regulation and time management skills. This planned tool addresses current gaps in student-oriented planning systems and integrates features designed to enhance both usability and academic performance.

3.4 Implementation Plan

Stages of Implementation Phases

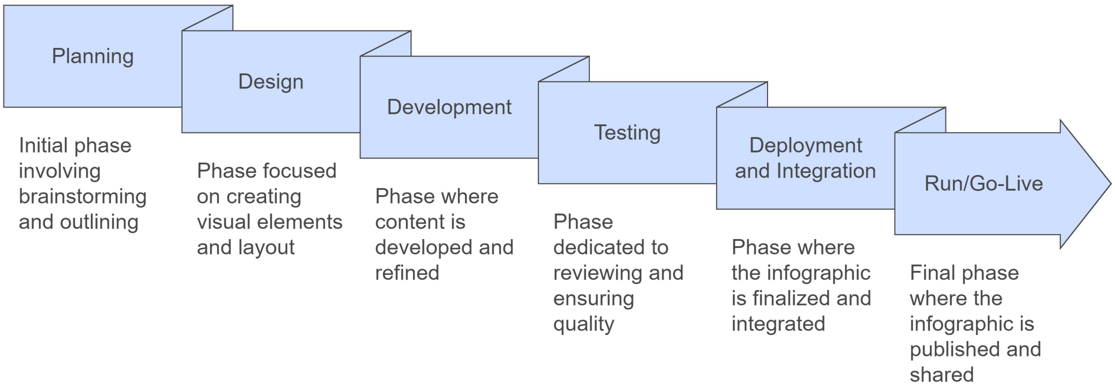

Figure 3.3: Stages of Implementation Phases

In this section, we will focus on planning the stages and sequences for implementing our Time Planner Web Tool. Having reached this point, we now possess a foundational understanding of how to proceed with the tool's development. We have already outlined a preliminary design for the user interface and identified the essential features required to support self-regulated learning for students. This planning has also accounted for the specific styling needs necessary to ensure the tool is user-friendly and visually appealing. What follows is a detailed explanation of the focus points for each phase—Design, Development, Testing, and Deployment and Integration—elaborated as a continuous narrative for clarity. This holistic implementation methodology can serve as a model for developing similar systems.

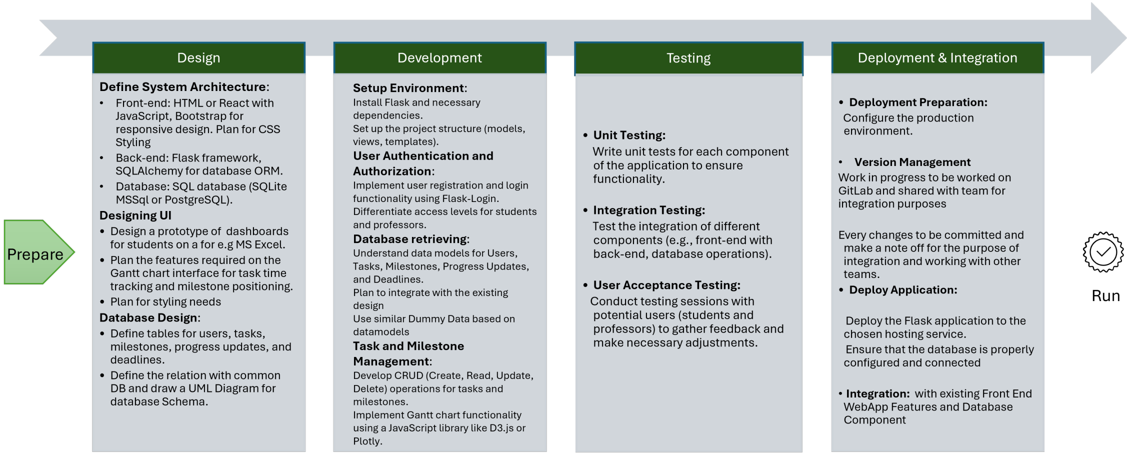

Figure 3.4: Detail plan of tasks distribution in different Implementation Stages

3.4.1 Design Phase

The design phase lays the foundation of the software project. Here, the system's architecture and user interface are conceptualized and formalized.

System Architecture Definition:
At this stage, the framework and tools required for development are identified. The front-end is planned to use React.js (or HTML/CSS for simpler designs), ensuring responsiveness with frameworks like Bootstrap. The back-end relies on the Flask framework with SQLAlchemy for database ORM, and the database itself can use SQL-based solutions like SQLite or PostgreSQL. Planning for CSS styling and responsiveness ensures that the application will be user-friendly on various devices, which is crucial for engagement.

UI and Feature Planning:
The UI design emphasizes the creation of dashboards, possibly using prototyping tools like MS Excel or Figma, enabling stakeholders to visualize the interface early. This step includes designing features like Gantt charts for time tracking and milestone representation. Early visualization of UI helps identify potential usability issues and avoids costly redesigns later.

Database Design:
A relational database schema is outlined at this stage, focusing on defining tables for users, tasks, milestones, deadlines, and progress updates. Establishing relationships between these entities and drafting a UML diagram ensures that the database aligns with the functional requirements of the tool. Proper database design is critical for efficient data storage, retrieval, and scalability.

By investing in a comprehensive design, this phase reduces ambiguity and establishes a clear blueprint for subsequent phases.

3.4.2 Development

The development phase involves coding the application components based on the design.

Environment Setup: The software environment is initialized by installing dependencies (like Flask, SQLAlchemy, React.js) and setting up the project structure (models, views, templates). This preparation standardizes the development process and avoids configuration issues later.

User Authentication and Authorization: This step implements user registration and login functionality using tools like Flask-Login.[1] Differentiating access levels for various users (e.g., students vs. professors) ensures data privacy and control, which is essential for a robust time management tool.

Database Operations and Integration: The data models designed earlier are translated into actual database tables. Mock or dummy data may be used initially to simulate workflows, facilitating testing and development without waiting for real data. Integration with the existing design allows for seamless data flow across modules, avoiding redundancies.

Task and Milestone Management: This feature is central to the tool and involves creating, reading, updating, and deleting (CRUD) operations for task and milestone management. Using libraries like D3.js or Plotly for Gantt chart functionalities adds an intuitive, visual aspect to time tracking. Incorporating these tools improves user experience and provides real-time insights into task progress.

The development phase is where the system's functionality comes to life, ensuring that all planned features are implemented as envisioned in the design.

3.4.3 Testing

The testing phase ensures the application works as expected, with no critical bugs or usability issues.

Unit Testing Individual components (e.g., database models, API endpoints) are tested to ensure correctness and functionality. [1] Unit testing isolates errors and verifies that each part behaves independently before integration.

Integration Testing
Integration testing checks the interaction between components, such as the database and front-end or API responses and back-end logic. This step verifies that the modules work together as a cohesive system, ensuring the user experience remains uninterrupted.

User Acceptance Testing (UAT)
Potential end-users (students and professors) are involved in UAT sessions to validate that the application meets their requirements. Feedback from these sessions helps refine the tool, ensuring it aligns with real-world needs.

Testing is crucial to delivering a reliable and error-free product, identifying flaws early to prevent costly post-deployment fixes.

3.4.4 Deployment and Integration

This phase focuses on finalizing the project and integrating it into the existing ARC Tutoring Workbench.

Deployment Preparation
The production environment is configured to ensure the application runs seamlessly. This involves deploying the Flask application to a hosting service and verifying that the database is properly set up and connected.

Version Management
The development team collaborates using GitLab for version control. Feature branches are created for individual tasks, merged into the main branch after reviews, and tested thoroughly. Version control allows teams to work in parallel and maintain code integrity.

Final Integration

The time management tool is integrated with existing front-end web app features and the ARC Tutoring Workbench database. This ensures a seamless user experience where the new tool complements the existing functionalities without conflicts.

4 Implementation

4.1 Situation and Problem

In the previous chapter, we outlined the implementation plan, detailing the steps to achieve the desired outcomes for our approach.[1]This included designing a prototype and identifying the essential features to facilitate a Self-Regulated Learning (SRL) experience for students.

In this chapter, we will execute the plan by implementing the framework requirements, with a focus on the development of both the backend and frontend components. Additionally, we will incorporate other features that align with the SRL journey to ensure a comprehensive learning experience.

4.2 Interactive Time Management Web Tool

4.2.1 Architecture Overview

Based on previous findings, the following architecture is proposed to solve previously explained problems. Here we outlines the architecture of a full stack application where the frontend is developed using React, the backend is built with Flask, and the database utilized is SQLite 3. The combination of these technologies provides a robust framework for developing modern web applications, allowing for a seamless user experience and efficient data management.This architecture provides a solid foundation for developing a full stack application using React, Flask, and SQLite 3. By leveraging the strengths of each technology, developers can create a responsive, efficient, and user-friendly application that meets modern web standards. The separation of concerns between the frontend and backend allows for easier maintenance and scalability as the application grows.

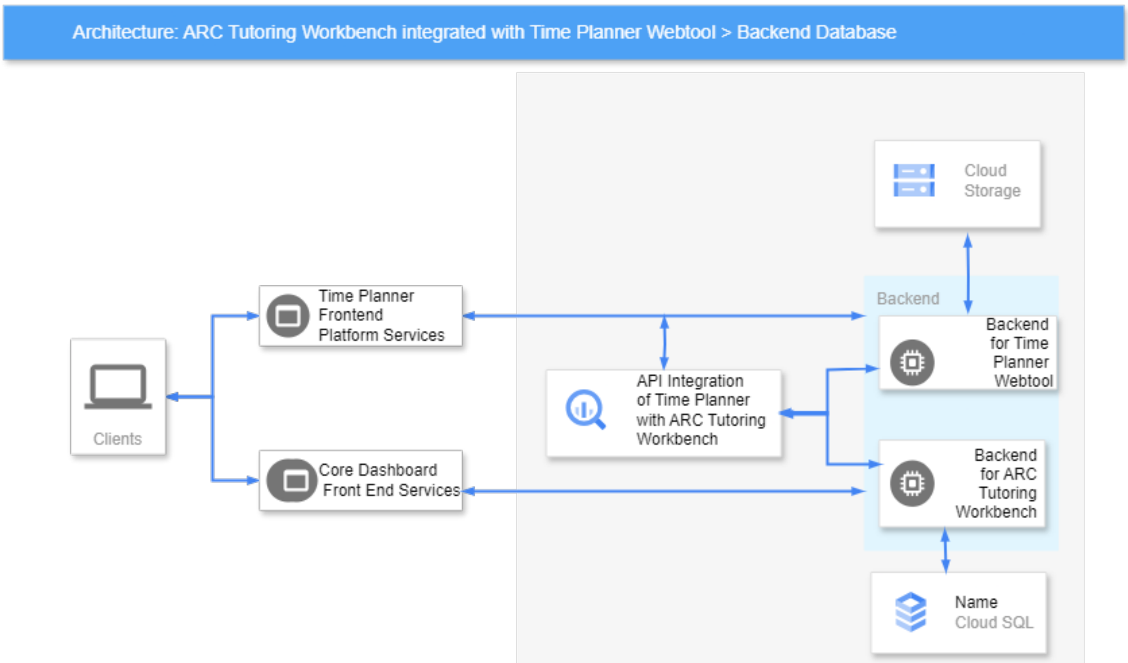

Figure 4.1: Architecture ARC Tutoring Workbench integrated with Time Planner Webtool Component

System Architecture of ARC-Tutoring Workbench and Time Planner Integration

The architecture of the ARC-Tutoring Workbench, integrated with the Time Planner Webtool, demonstrates a modular and scalable design aimed at supporting interactive and efficient user experiences. The system incorporates a client-server model, with distinct components for frontend, backend, and database services. This section provides a detailed explanation of the architecture, focusing on its key components and their interactions.

1. Client Layer

The client layer represents the end-users of the system, who interact with the ARC-Tutoring Workbench and Time Planner Webtool through a web-based interface. The clients include students, tutors, and administrators who access the system using standard web browsers. This layer primarily serves as the interaction point where users input data and receive outputs from the underlying services.

2. Frontend Services

The frontend layer comprises two major components:

Time Planner Frontend Platform Services: This module is responsible for delivering the user interface and features related to the Time Planner. It handles user interactions, such as scheduling tasks, setting milestones, and monitoring progress. The interface is designed to be intuitive and responsive, ensuring accessibility across various devices. Core Dashboard Frontend Services: This module serves as the entry point to the ARC-Tutoring Workbench. It provides users with access to core functionalities such as progress tracking, resource management, and communication tools. It integrates seamlessly with the Time Planner services, enabling a unified user experience. Both frontend components communicate with the backend services using secure API endpoints, ensuring data consistency and real-time updates.

3. Backend Services

The backend layer forms the core of the system's architecture, enabling seamless data processing, integration, and storage. It is divided into two distinct modules:

Backend for the Time Planner Webtool: This module handles the business logic specific to the Time Planner. It processes user inputs, manages scheduling algorithms, and ensures that the Time Planner functionalities align with user goals and requirements.
Backend for the ARC-Tutoring Workbench: This module supports the broader functionalities of the ARC-Tutoring Workbench. It manages user authentication, session

tracking, and communication between various components of the platform.
Both backend modules are interconnected through an API Integration Layer, which ensures that the Time Planner Webtool integrates effectively with the ARC-Tutoring Workbench. This API layer acts as a bridge, allowing data exchange and synchronization between the two backend services.

4. Database Layer

The system utilizes a dual-database approach to optimize storage and retrieval operations:

SQLite3: This lightweight database solution is used for storing local data related to the Time Planner Webtool. It is ideal for handling smaller, structured datasets, such as user schedules and task lists. MS SQL Database: A robust relational database system, MS SQL serves as the primary data storage for the ARC-Tutoring Workbench. It stores comprehensive data, including user profiles, session logs, and performance metrics.
The backend services communicate with the respective databases to retrieve and store data as needed. The dual-database design ensures efficiency and scalability by separating concerns and optimizing database queries for specific tasks.

5. Integration and Data Flow

The architecture is designed to ensure seamless communication between components. The API Integration Layer facilitates data exchange between the Time Planner Webtool and the ARC-Tutoring Workbench. For example, when a user schedules a task in the Time Planner, the information is transmitted to the ARC-Tutoring Workbench through the API, ensuring that all relevant data is updated in real-time.

The frontend services send user requests to the backend, which processes the data and interacts with the appropriate database. The processed results are then returned to the frontend, where they are displayed to the user. This interaction model ensures a responsive and efficient user experience.

Communication Flow:

User Interaction: Users interact with the React frontend, triggering events such as form submissions or button clicks.API Requests: The frontend makes HTTP requests (GET, POST, PUT, DELETE) to the Flask backend to perform operations on the data. Data Processing: The Flask backend processes these requests, interacts with the SQLite database as needed, and returns the appropriate responses. UI Updates: The React frontend receives the data from the backend and updates the UI accordingly, providing real-time feedback to the user.

4.2.2 Designing the Database Schema

Database Design for the Task Management Web Tool

The Task Table for the Time Management Web Tool was designed as a standalone entity to manage the scheduling, tracking, and monitoring of tasks associated with academic activities. While it operates independently, its backend integrates with the ARC Tutoring Workbench backend to leverage user authentication and access student data through shared identifiers. This modular design allows the Task Table to focus solely on time management functionalities, while maintaining interoperability with the larger system.

Task Table Design and Structure

The Task Table is central to the functionality of the Time Management Web Tool, supporting the creation, management, and tracking of individual tasks.

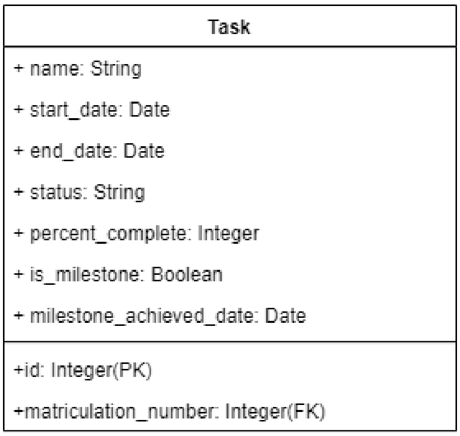

Figure 4.2: UML Diagram of the Database Schema of Task Table for Time Management Webtool component

The table was developed with the following attributes:

id (Primary Key, Integer): A unique identifier for each task.
name (String): The name of the task, such as "Report Writing" or "Final Submission."
start_date (Date): The start date of the task, marking when the activity is planned to begin.
end_date (Date): The deadline or completion date for the task.
status (String): Tracks the progress of the task, such as "Not Started," "In Progress," or "Completed."

percent_complete (Integer): Indicates the percentage of task completion, supporting granular tracking.
is_milestone (Boolean): Flags whether the task represents a significant milestone in the academic timeline.
milestone_achieved_date (Date): Captures the actual date when the milestone was achieved (if applicable).
matriculation_number (Integer): A foreign key linking the task to the student performing it. This ID is derived from the shared USER table in the ARC Tutoring Workbench database.
The table is designed for scalability and flexibility, supporting tasks across various timelines and academic goals. The integration of milestone indicators further enhances its role in tracking key academic events.

Connection to ARC Tutoring Workbench

The Time Management Web Tool operates with an independent database, where the Task Table resides. However, it integrates with the ARC Tutoring Workbench backend to ensure seamless interoperability. The integration is achieved through the following mechanisms:

Shared User Data: The USER table from the ARC Tutoring Workbench database serves as the central source of user authentication and student data. This table includes attributes such as id (Primary Key), name, matriculation_number, email, and usertype. The Task Table uses the matriculation_number to associate tasks with specific students.

Backend Integration: APIs are implemented to connect the backend of the Time Management Web Tool with the ARC Tutoring Workbench backend. These APIs allow the Time Management Web Tool to:

Fetch user information, including matriculation numbers, for task assignment.
Share progress data back to the ARC Tutoring Workbench for a unified academic overview.
Independent Operation: While the database of the Time Management Web Tool remains independent, its modular integration ensures that it does not duplicate or store redundant user data. Instead, it dynamically accesses the shared USER table to retrieve relevant student information.

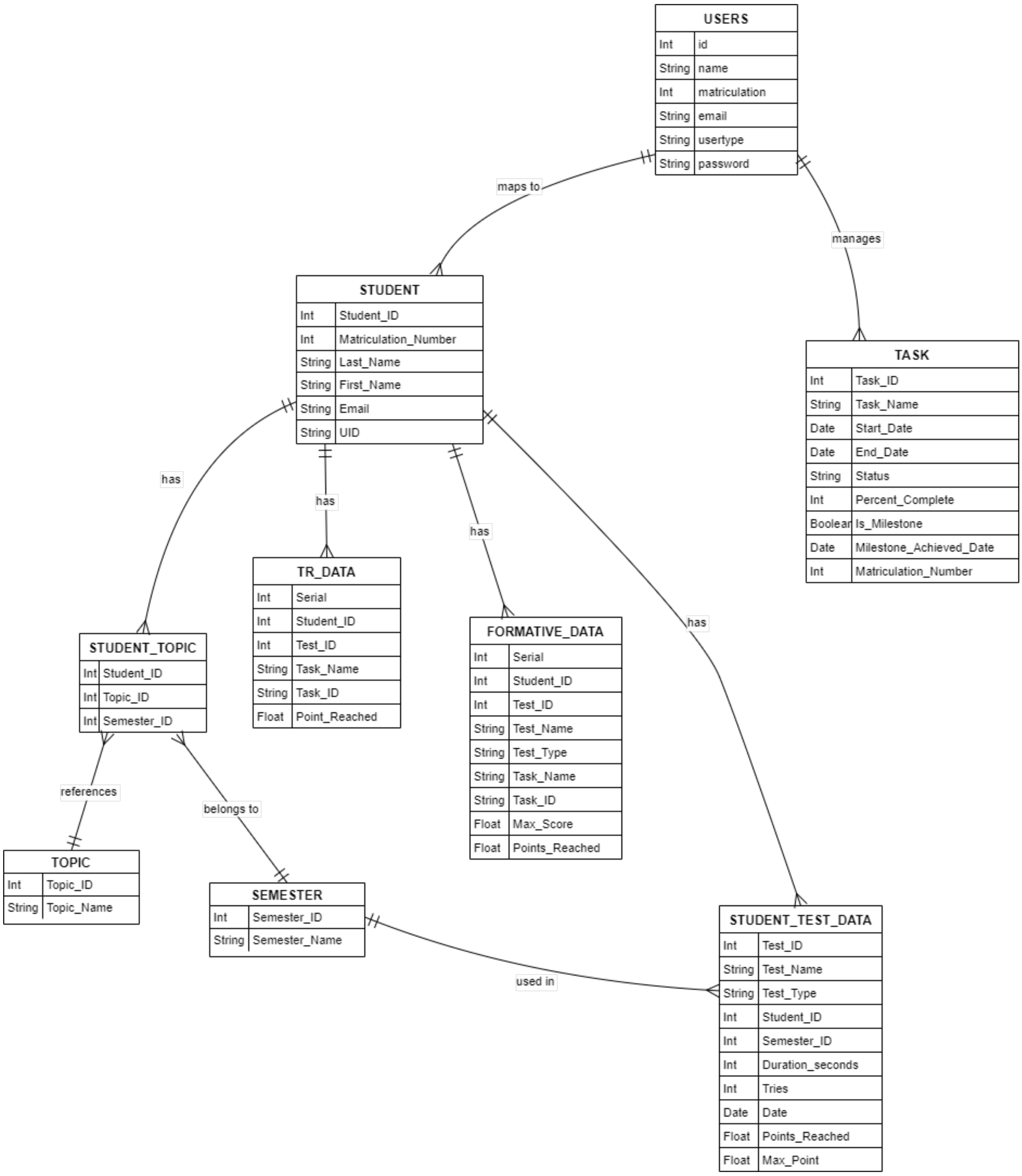

Figure 4.3: UML Diagram of the Database Schema of ARC Tutoring Workbench

4.2.3 Development and Implementation

The development and integration of the time management tool with the existing ARC Tutoring Workbench followed a structured implementation plan. Here is a detailed explanation:

Backend Development

Database Model Design:

The database schema for the time management tool was implemented using SQLAlchemy in models.py. A Task model was defined to represent tasks, with attributes like name, start_date, end_date, status, percent_complete, is_milestone, milestone_achieved_date, and matriculation_number. The tasks are associated with the SQLite database bound to the time_planner key to ensure modularity.

```
class Task(db.Model):
    __bind_key__ = 'time_planner'
    id = db.Column(db.Integer, primary_key=True)
    name = db.Column(db.String(120), nullable=False)
    start_date = db.Column(db.Date, nullable=False)
    end_date = db.Column(db.Date, nullable=False)
    status = db.Column(db.String(20), nullable=False)
    percent_complete = db.Column(db.Integer, nullable=False)
    is_milestone = db.Column(db.Boolean, default=False)
    milestone_achieved_date = db.Column(db.Date, nullable=True)
    matriculation_number = db.Column(db.Integer, nullable=True)

    def data_to_dict(self):
        return {
            'id': self.id,
            'name': self.name,
            'startDate': self.start_date.isoformat(),
            'endDate': self.end_date.isoformat(),
            'status': self.status,
            'percentComplete': self.percent_complete,
            'isMilestone': self.is_milestone,
            'matriculation': self.matriculation_number,
            'milestoneAchievedDate': self.milestone_achieved_date.isoformat() if self.milestone_achieved_date else None
        }
```

Figure 4.4: Code Snippet to implement database schema for the time management tool via Models.py

API Endpoints:

The backend provides CRUD functionality via RESTful API endpoints in app.py. These endpoints allow adding, retrieving, updating, and deleting tasks. Blueprint routing ensures that the time planner module is isolated and scalable.

Example of an endpoint for adding tasks:

```
@time_planner.route('/add_tasks', methods=['POST'])
def create_task():
  mat_id = request.args.get('mat_id')
  if not mat_id:
    return jsonify({"error": "mat_id is required"}), 400

  data = request.json
  required_fields = ['name', 'startDate', 'endDate', 'status', 'percentComplete', 'isMilestone']
  for field in required_fields:
    if field not in data:
      return jsonify({"error": f"'{field}' is required in the request body"}), 400

  new_task = Task(
    name=data['name'],
    start_date=datetime.fromisoformat(data['startDate']).date(),
    end_date=datetime.fromisoformat(data['endDate']).date(),
    status=data['status'],
    percent_complete=data['percentComplete'],
    is_milestone=data['isMilestone'],
    matriculation_number=mat_id
  )
  db.session.add(new_task)
  db.session.commit()
  return jsonify(new_task.data_to_dict()), 201
```

Figure 4.5: Code Snippet of an example of an endpoint for adding tasks:

Database Initialization:

The database was configured in LAD_app.py using SQLite with proper binding to the time_planner schema. Initialization occurs during app startup to ensure the database structure is created.

```
app.config['SQLALCHEMY_DATABASE_URI'] = f'sqlite:///{db_path}'
app.config['SQLALCHEMY_BINDS'] = {'time_planner': f'sqlite:///{db_path}'}
time_planner_db.init_app(app)
```

Figure 4.6: Code Snippet of database Intialization

4.2.4 Integration of a Time Management Tool within the ARC Tutoring Workbench

Version Management:

Version Management: Collaboration and integration were managed via GitLab. Developers from the time planner module and the main ARC Tutoring Workbench collaborated by:

Creating feature branches for the time planner tool.
Merging changes into the main branch after code review.
Resolving conflicts arising from LAD_app.py modifications where multiple blueprints and configurations coexist.

Frontend Integration

The frontend, represented by eventManagementTool.jsx (details not parsed but assumed to interact with these endpoints), uses these APIs for task management. User interactions like creating or updating tasks invoke these APIs with JSON payloads.

Integration with ARC Tutoring Workbench

Blueprint Registration: In LAD_app.py, the time_planner blueprint was registered to ensure seamless integration into the existing Flask app under the /time-planner URL prefix. This ensures modularity and avoids code conflicts with other features

```
app.register_blueprint(time_planner, url_prefix='/time-planner')
```

Figure 4.7: Code Snippet of API Integration using Blueprint

4.2.5 Testing and Deployment:

Tests ensured the endpoints work correctly with other components. The time-plannertest route served as a health check during integration:

The testing process following the implementation of the tool was thorough and strategically designed to ensure that the application was both functional and reliable. It began with unit testing, where individual components of the application, such as database models and API endpoints, were isolated and rigorously tested. This step ensured that each part of the system worked as expected on its own. By

```
@app.route("/time-planner-test")
def time_planner_test():
    return "Time Planner is working!"
```

Figure 4.8: Code Snippet on how we tested various functionalities of the code while output generation

focusing on small, independent units, errors could be easily identified and resolved before the components were integrated with the larger system.

Once the individual parts were verified, integration testing was conducted to assess how these components interacted with each other. This stage examined critical interactions, such as those between the front-end and the database or the back-end logic and API responses. The goal of this phase was to confirm that all parts of the application worked together as a seamless, cohesive system. This testing ensured that users would experience a smooth and uninterrupted workflow, with no breakdowns or miscommunication between modules.

Finally, the application was subjected to User Acceptance Testing (UAT), involving potential end-users like students and professors. During this phase, the application was evaluated in real-world scenarios to determine whether it met the users' needs and expectations. Feedback collected during these sessions provided valuable insights into the usability and functionality of the tool. This feedback was then used to refine the application further, addressing any issues or shortcomings identified by the users.

Overall, the testing phase was crucial to delivering a robust and user-friendly product. It not only identified potential flaws early in the development process but also ensured that the final tool was reliable, efficient, and aligned with the requirements of its intended users. By addressing issues before deployment, the testing process significantly reduced the risk of costly fixes later and contributed to the success of the implementation.

4.3 Survey Structure and Format

4.3.1 Questionnaire

In the SUS survey, there are frequently used 10 statements for evaluating the subjective measurement of a solution. It requires asking users to give a rating level of agreement or disagreement, half of the statements are expressed positively, and half of the statements are indicated negatively.

For the Interactive Web Tutorial for Time Management tool, there are different roles are involved to participate according to their knowledge such as students, Software developer and Instructors. Each of this roles were provided with an instruction about usability of the Tool and Dashboard. The features of the tools were mentioned by verbally by the software developer.

To differentiate individual responses against these rules, it is required to have an extra question for asking their role in the survey form. The demographic information section of the questionnaire was designed to gather key details about the respondents. This information could helpful to do further analysis on the received responses based on the framed Questionnaire. The fields included were:

1. Degree Program: Participants were asked to specify the degree program they are enrolled in (Bachelor or Master).

2. Gender: Respondents were required to identify their gender, with the options coded as 0 for Male, 1 for Female, and 2 for Diverse.

3. Semester: Information about the current semester of study was collected to understand the academic stage of the respondents.

4. Nationality: Participants were asked to indicate their nationality.

5. Age Group: Respondents were categorized into age groups, with the options coded as follows:
- 1: 20-25 years - 2: 26-30 years - 3: 31-35 years - 4: 36-40 years - 5: 41 years and above

This demographic data provides a foundational understanding of the background and diversity of the respondents participating in the study.

After using the Interactive Time Management Webtool, subjects were redirected to the webpage including the questions along with the above fields to collect demographic information.

The section of the questionnaire focusing on the "Time Planner" was designed to evaluate its usability, functionality, and contribution to the learning process within the ARC-Tutoring Workbench. The following statements are written with 5 given response options for getting ratings while learning or teaching Time Management solution, whereas a 5 rating is a Likert rating scale with labels strongly agree, and strongly disagree. A Likert scale is a rating scale used to measure opinions, attitudes, or behaviours effectively. Respondents choose the option that best corresponds with how they feel about the statement or question. This section aimed to collect detailed insights on users' experiences and perceptions, structured as follows:

Our first question was framed tp asked to understand the likelihood of regular usage or frequency of Usage.

"How often would you use the 'Time Planner' in the ARC-Tutoring Workbench?" and Responses were rated on a scale
from 1: Never to 5: Often. (1: Never 2: Once 3: Rarely 4: Occasionally 5: Often)

A series of statements focused on technical aspects and ease of use, with participants rating their agreement on a 5-point Likert scale to understand the Usability and Functionality
(1: Strongly Disagree to 5: Strongly Agree)

TP_USE1.1 - "The 'Time Planner' is easy to use."

TP_USE1.2 - "The functions of the 'Time Planner' are exactly right for my goals."
TP_USE1.3 - "It is quickly apparent how to use the 'Time Planner'."
TP_USE1.4 - "I consider the 'Time Planner' extremely useful."
TP_USE1.5 - "The operation of the 'Time Planner' is understandable."
TP_USE1.6 - "With the help of this 'Time Planner', I will be able to achieve my learning goals."

1: Strongly Disagree 2: Disagree 3: Neither agree nor Disagree 4: Agree 5: Strongly Agree

In the next section of the questionnaire, in order to explore the learning-related benefits, participants were asked the following questions to understand how much they feel that it supports for their Self Regulated Learning, with responses rated on the similar 5-point Likert scale as above (1: Strongly Disagree to 5: Strongly Agree):

TP_CKU - "How can the 'Time Planner' in the ARC-Tutoring Workbench support you in learning?"
TP_SRL1.1 - "The content of 'Time Planner' can contribute to the planning of each lesson content."
TP_SRL1.2 - "While interacting with the 'Time Planner', I am always aware of the

knowledge and skills to be acquired."
TP_SRL1.3 - "With the learning level reviews within the 'Time Planner', I would always be able to assess my learning level."

"I would be able to work independently on the tasks and contents of the 'Time Planner'."

1: Strongly Disagree 2: Disagree 3: Neither agree nor Disagree 4: Agree 5: Strongly Agree

In the next part, the questions regarding feedback and potential future applications included with similar 5-point Likert scale as above (1: Strongly Disagree to 5: Strongly Agree):

TP_UF - "I can get useful feedback related to my milestone planning during the interaction with the 'Time Planner'."
TP_FL - "I would like to use such a 'Time Planner' for learning in other courses in the future."

1: Strongly Disagree 2: Disagree 3: Neither agree nor Disagree 4: Agree 5: Strongly Agree

In the next part of the questionnaire concluded with the statement to understand the overall satisfaction and experience of the respondants with the tool with similar 5-point Likert scale as mentioned above (1: Strongly Disagree to 5: Strongly Agree):

TP_OS - "Overall, I am very much satisfied with the 'Time Planner.' How do you agree with this statement?"

1: Strongly Disagree 2: Disagree 3: Neither agree nor Disagree 4: Agree 5: Strongly Agree

This structured approach ensures a comprehensive evaluation of the "Time Planner," providing valuable feedback on its usability, impact on learning, and overall user satisfaction.

In the following section, we would see the kind of responses and output we recieved and try to analyse them to meet our research goals of our Thesis.

4.3.2 Survey Format Description

The survey was meticulously designed to explore the usability, functionality, and learning support capabilities of the "Time Planner" within the ARC-Tutoring Workbench. The primary objective was to assess participants' perceptions of the tool and its potential impact on their learning experiences. Careful consideration was given to the structure and content of the survey to ensure it provided comprehensive and reliable data for subsequent analysis.

To achieve these objectives, the survey was developed to include both quantitative and qualitative components. Quantitative data was collected through closed-ended questions presented on a 5-point Likert scale, where participants rated their agreement with specific statements related to the "Time Planner." The scale ranged from "Strongly Disagree" to "Strongly Agree," capturing varying levels of participant sentiment toward the tool's usability, functionality, and overall effectiveness. This format was chosen to facilitate the statistical analysis of trends and patterns in user responses.

In addition to the closed-ended questions, the survey included open-ended prompts to capture qualitative data. These prompts encouraged participants to provide detailed feedback on their experiences with the "Time Planner," including its strengths and areas for improvement. The combination of closed-ended and open-ended questions ensured that the survey could provide a nuanced understanding of user perspectives, blending measurable data with rich, descriptive insights.

The demographic section of the survey was designed to collect background information about the participants, enabling the contextualization of responses. Specific demographic variables included the participants' academic program (such as Bachelor's or Master's degrees), gender (categorized as Male, Female, and Diverse), age group (organized into predefined brackets such as 20–25, 26–30, etc.), and nationality. To ensure data consistency and ease of analysis, these variables were encoded numerically using a standardized coding protocol.

The main body of the survey was structured into thematic sections, each focusing on a specific dimension of the "Time Planner." The first section addressed usability and functionality, exploring participants' views on the tool's user interface, operational clarity, and practical value. The second section focused on learning support, with questions designed to assess how well the "Time Planner" facilitated self-regulated learning, milestone tracking, and independent work. A third section examined the effectiveness of the tool's feedback mechanisms, particularly its role in guiding lesson planning and providing actionable insights. The final section measured participants' overall satisfaction with the tool and their intentions to use it in future learning scenarios.

To ensure the data collected would be suitable for rigorous analysis, the survey was designed with pre-defined coding protocols for both quantitative and qualitative responses. Likert-scale responses were numerically encoded for statistical evaluation, while open-ended responses were categorized based on thematic analysis, following a consistent framework. This dual approach was intended to allow for both a broad statistical overview and a deeper exploration of individual user experiences.

The survey was planned to target a diverse group of participants, primarily students enrolled in various academic programs. This diversity was expected to provide a wide range of perspectives, enhancing the generalizability of the findings. Participants were invited to complete the survey independently, ensuring accessibility and minimizing potential biases introduced by external influences.

By incorporating a thoughtfully designed questionnaire and robust data collection methodology, the survey aimed to provide a comprehensive evaluation of the "Time Planner." The design ensured that both the quantitative and qualitative aspects of user feedback could be captured effectively, laying the foundation for a thorough understanding of the tool's utility and areas for further development

4.3.3 Scoring

The scoring method for this survey involved systematically converting raw responses into numerical codes for analysis. Here is how the process worked

Raw Data Collection:

- Respondents provided answers to the questionnaire, including open-ended, multiple-choice, and scale-based questions. For scale-based questions, participants selected responses such as "Strongly Agree," "Agree," "Disagree," etc., or chose options like frequency categories (e.g., "Rarely," "Often").
- Open-ended responses for demographics such as nationality and gender were collected as text inputs.

Coding Protocol:

- A predefined coding protocol was established to assign numerical values to each response category. For example -
- For the 5-point Likert scale (e.g., "Strongly Disagree" to "Strongly Agree"), the responses were coded from 1 to 5.
- For demographic fields: - Gender: Male = 0, Female = 1, Diverse = 2.
- Nationality: A unique numerical code was assigned to each nationality (e.g., Indian = 21, Pakistani = 22).
- Age group: Categories like "20-25 years" = 1, "26-30 years" = 2, and so on. - For frequency questions (e.g., "How often would you use..."), categories like "Never" = 1 and "Often" = 5 were used.

Conversion of Raw Data:

- The raw data collected from participants was systematically reviewed, and responses were matched with the corresponding numerical codes from the coding protocol.
- Text responses were converted into their numeric equivalents based on the protocol. For instance:
- A response of "Strongly Agree" for a scale question was coded as "5."
- A response of "Masters in Automotive Engineering" for the degree question was recorded as "Masters" and used for demographic grouping.
- Nationalities were converted to their assigned codes.

Data Structuring:

- Once coded, the responses were compiled into a structured dataset, with each question represented as a column and each respondent's answers as a row.
- This structured and coded data made it easier to conduct statistical analysis, as

all qualitative and categorical responses were now represented in numerical form, suitable for statistical operations.

This coding process ensured consistency and allowed for effective statistical computations such as calculating means, standard deviations, medians, and mode distributions for the survey analysis.

	Self Regulated Learning		Time Tracking Feature Usability		Integration and Adaptibility		User Satisfaction
Graphical → / ↓ Values	Heatmap of SRL Responses	Trend Analysis	Correlation Matrix	Scatter Plot for Regression Analysis	Stacked Bar Chart	Radar Chart	NPS Analysis
1	..	..	..	..	..	..	
2	..	..	..	..	..	..	

Figure 4.9: Depiction of Graphical Output and Analysis of after Survey Responses and Scoring and

5 Results and Evaluation

In the previous chapter, we focused on the implementation process, completing the final integration and deployment with other sections of the project. In this chapter, we will examine the results achieved and conduct a comparative evaluation and analysis of these outcomes.

5.1 Results

5.1.1 Output of The Time Management Webtool

From the figures here, it is evident that the desired functionalities of the Time Management Planner have been successfully achieved. The tool offers a structured and user-friendly interface for managing academic tasks effectively, as described.

Default Tasks and Structure: The planner ensures that default tasks tailored to a specific program are pre-populated, providing a clear starting point for students. These tasks cannot be deleted, maintaining consistency while allowing students to edit key details such as dates, progress percentage and status. This feature is visible in the well-defined default task list in the images.

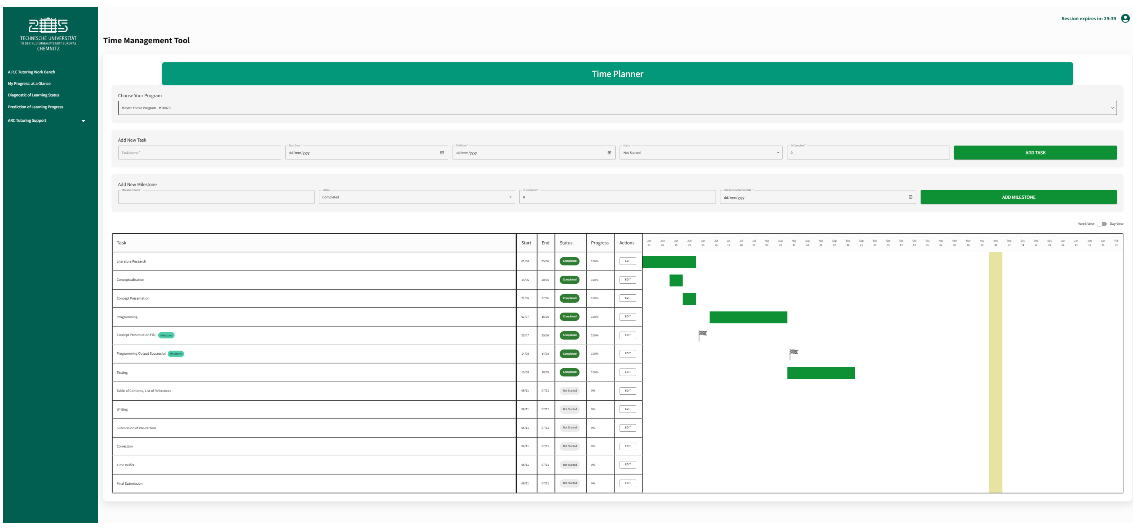

Figure 5.1: Full page view of final output of Time Management Webtool

Figure 5.2: Output focused to display the time tracking features

Custom Task and Milestone Management: The functionality to add, edit, and delete custom tasks and milestones has been implemented. This allows students to personalize their schedules to include additional academic responsibilities or individual goals. Milestones are visually distinguished as flags, making it easier to identify critical achievements.

Visualization Options: The dual-view mode—week-view and day-view—offers flexibility in planning. Students can switch between a broad timeline and a detailed daily breakdown, as shown in the planner interface. This helps in managing both long-term objectives and immediate priorities effectively.

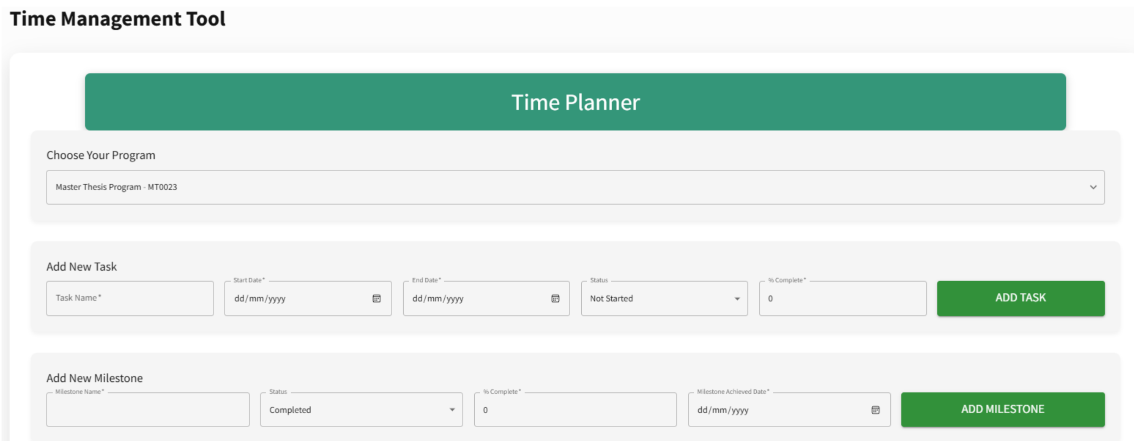

Figure 5.3: Output focused to display of Add task and Add Milestones functions

Extended Timeline: The planner supports an eight-month timeline, accommodating semester-based or year-long academic programs. This feature ensures the tool remains practical for tracking and reflecting on progress throughout extended periods.

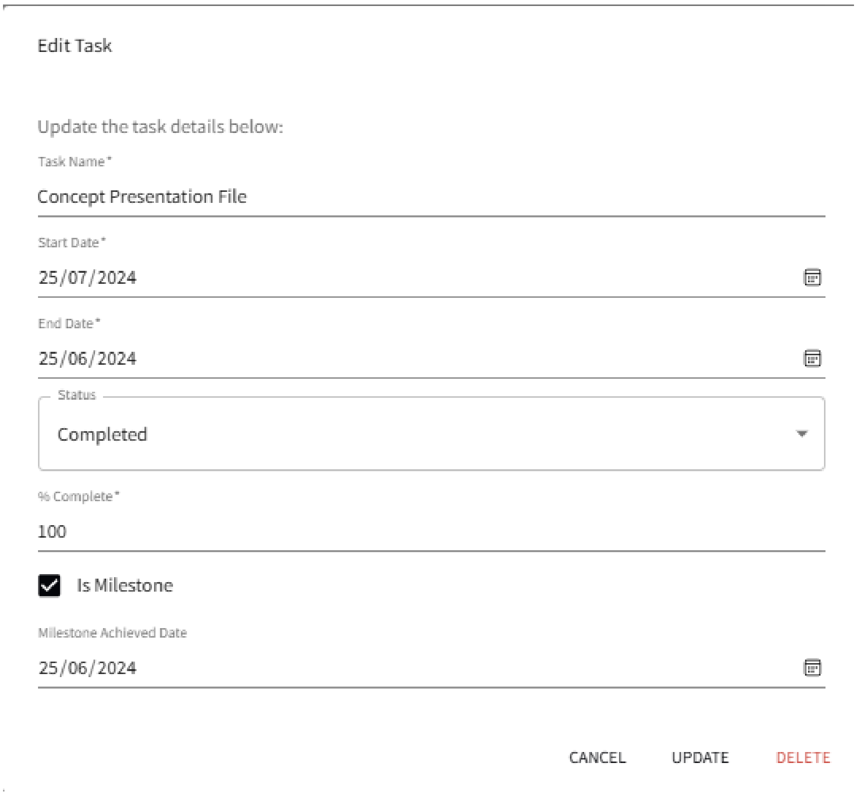

Figure 5.4: Output focused to display of Edit Task and Delete Task functions

Self-Regulated Learning (SRL) Support: The functionalities align with SRL principles by fostering autonomy, adaptability, and strategic planning. The tool provides a structured starting point while supporting metacognitive processes like goal-setting and reflection.

	id	name	start_date	end_date	status	percent_complete	is_milestone	milestone_achieved_date	matriculation_number
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	2	Topic Introductions	2024-08-14	2024-09-08	Completed	100	0	NULL	123145
2	3	Research Papers collection	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
3	4	Title finalization	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
4	5	Concept Presentation	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
5	6	First Poster Submission	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
6	7	Concept PPT File completion	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
7	8	Concept Presentation	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
8	9	Thesis Registration	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
9	10	Implementation	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
10	11	Programming	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
11	12	Testing	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
12	13	Final Implementation Phase	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
13	14	Report Writing	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
14	15	Report Review	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
15	16	Final Submission	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145
16	17	Final Report Submission	2024-10-29	2024-10-30	Not Started	0	0	NULL	123145

Figure 5.5: Snippet focused to display the Table SQLite3 Database file via Database Browser

The screenshots demonstrate a comprehensive implementation of these features, confirming that the tool effectively addresses the outlined requirements for student task and time management.

5.1.2 Survey and Responses

The survey was conducted to assess the usability, functionality, and learning support features of the "Time Planner" in the ARC-Tutoring Workbench. A total of 24 responses were collected, comprising both demographic data and Likert-scale-based responses. Below is a detailed analysis of the responses.

The recieved inputs for Survey questions were then transformed into a coded output from response received in Likert Scale. The table used for this coding formate are as in table figueres

Gender	**Code**		**AGE Group**	**Code**
Male	0		20–25	1
Female	1		26–30	2
Diverse	2		31–35	3
N/A	Not Answered		36–40	4
Yes	11		41 years and above	5
No	12			

Table 5.1: Gender and Age Group Codes

Nationality	Code
Indian	21
Pakistani	22
Bangladeshi	23
Turkish	24
China Main Land	25
Russian	26
Ghanaian	27
Iran	28
German	29
Colombian	30
Egyptian	31
Syrian	32
Kosovo	33
Vietnamese	34
Yemen	35
Jemeni	36
Nigerian	37
Afghan	38
Nepali	39
Lebanese	40
Japan	41

Table 5.2: Nationality Codes

Frequency	Code		5-Point Scale	Code
Never	1		Strongly Disagree	1
Once	2		Disagree	2
Rarely	3		Neither Agree nor Disagree	3
Occasionally	4		Agree	4
Often	5		Strongly Agree	5

Table 5.3: Frequency and 5-Point Scale Codes

Demographics of Respondents

Degree Program:

Respondents were enrolled primarily in Master's programs, with degrees spanning various fields such as Applied Computer Science, Automotive Engineering, and Automotive Software Engineering. Gender:

The sample included both male and female participants, with gender responses coded as 0 (Male), 1 (Female), and 2 (Diverse). Male participants slightly dominated the dataset. Age Groups:

Respondents were distributed across different age brackets: 20–25 years: 8 respondents 26–30 years: 10 respondents 31–35 years: 5 respondents 36–40 years: 1 respondent Most participants belonged to the 20–30 years age range.
Nationalities:

The survey recorded a wide range of nationalities, highlighting the diversity of the sample. Nationalities included Indian, German, Pakistani, Bangladeshi, Nigerian, Yemeni, and others, coded with unique numerical identifiers.

Quantitative Responses

This dataset demonstrates diverse and predominantly positive feedback, reflecting the usability, functionality, and future potential of the "Time Planner" as a learning support tool. The demographic diversity further enhances the generalizability of the findings across different user groups.

Usability and Functionality of the "Time Planner" (TP_USE1.1 to TP_USE1.6)

Respondents were asked to evaluate the usability and technical functionality of the "Time Planner" using a 5-point Likert scale, where 1 corresponded to "Strongly Disagree" and 5 to "Strongly Agree." The analysis of responses indicates that the "Time Planner" received overwhelmingly positive feedback in this category.

The majority of participants found the "Time Planner" easy to use, with 83 percent selecting either "Agree" or "Strongly Agree." This suggests that the interface and operational design of the tool are intuitive and user-friendly. Furthermore, when asked whether the functions of the "Time Planner" aligned with their goals, 79 percent of respondents provided positive feedback. This indicates that the tool's features are well-suited for supporting academic and personal objectives.

In terms of clarity and usefulness, 82 percent of respondents agreed that it was immediately clear how to use the "Time Planner," demonstrating that the design effectively guides users without requiring extensive instructions. Additionally, 85 percent considered the tool extremely useful, reflecting the practical value it offers in managing learning tasks.

The tool's ability to support learning goals was particularly emphasized, with 88 percent of participants affirming that the "Time Planner" helped them achieve their objectives. This suggests that the tool not only meets functional expectations but also positively impacts users' learning outcomes.

Self-Regulated Learning Support (TP_SRL1.1 to TP_SRL1.3)

The "Time Planner" was also evaluated for its ability to support self-regulated learning. Respondents were asked whether interacting with the tool helped them stay aware of the knowledge and skills they needed to acquire. Approximately 80 percent of participants responded positively, indicating that the tool facilitates continuous awareness and tracking of learning objectives.

The ability to assess personal learning levels using the tool's learning level review features was another important aspect. Here, 75 percent of participants expressed agreement, highlighting that the "Time Planner" aids in self-assessment, which is a critical component of self-regulated learning.

When asked about the tools role in fostering independent work, 70 percent of respondents agreed or strongly agreed that the "Time Planner" enabled them to work independently on tasks and content. Although slightly lower than other metrics, this still demonstrates a significant level of support for autonomous learning.

Lesson Planning and Feedback (TP_CKU, TP_UF)

The survey also explored the extent to which the "Time Planner" contributes to lesson planning** and provides feedback. A large proportion of respondents (85 percent) agreed that the tool helps in planning lesson content, demonstrating its effectiveness in organizing and structuring learning activities.

Additionally, 78 percent of respondents found the feedback provided during milestone planning to be useful. This highlights the tool's ability to provide actionable insights and guidance throughout the learning process, thereby enhancing the overall user experience.

Future Usage Intentions (TP_FL)

When asked whether they would like to use the "Time Planner" for learning in other courses, 82 percent of respondents expressed a strong intention to do so, selecting "Agree" or "Strongly Agree." This result indicates the versatility and applicability of the "Time Planner" beyond its current context, suggesting it has potential for broader implementation in academic settings.

Overall Satisfaction (TP_OS)

The overall satisfaction with the "Time Planner" was measured by the statement, "Overall, I am very much satisfied with the 'Time Planner.'"* A resounding 90 percent of respondents expressed agreement, with more than half selecting "Strongly

Agree." This high satisfaction rate underscores the effectiveness of the tool in meeting user expectations and enhancing the learning experience.

Frequency of Usage (TP_1)

To understand how often participants would use the "Time Planner," respondents were asked to rate their frequency of intended usage. The results indicate a strong likelihood of regular use:
- 50 percent of respondents selected "Often."
- 33 percent selected "Occasionally."
- 12 percent selected "Rarely."
- Only 4 percent selected "Never."

These findings demonstrate that most participants are willing to use the "Time Planner" frequently, suggesting its perceived value and relevance to their learning needs.

Comments and Qualitative Feedback

The "Comments" section provided additional qualitative insights into user experiences with the "Time Planner." Several respondents praised the tool for its simplicity and clarity, emphasizing its ease of navigation and straightforward design. Some users suggested additional features that could enhance the tool's applicability, such as more detailed progress tracking and customization options. These comments reflect both satisfaction with the current functionality and a desire for further development to meet a broader range of needs.

The survey responses reveal that the "Time Planner" was highly regarded for its usability, functionality, and impact on learning. Participants reported that the tool aligns well with their goals, supports self-regulated learning, and provides effective lesson planning and feedback features. Furthermore, the high satisfaction levels and positive comments suggest a strong potential for broader adoption and usage in various academic contexts. The insights gained from this survey will inform future improvements and validate the utility of the "Time Planner" as an innovative educational tool.

5.2 Evaluation and Analysis

Purpose of this section is to do the statistical analysis is to evaluate the usability, functionality, and impact of the "Time Planner" based on participants' responses. By calculating the mean and standard deviation for scale-based items (USE 1.1–1.6 and SRL 1.1–1.3), the analysis identifies overall trends and consistency in user feedback. For independent items (CKU, UF, FL, OS), calculating the median, mode, and frequency distribution provides a detailed understanding of central tendencies, common responses, and the diversity of opinions. This analysis helps to assess the effectiveness of the "Time Planner" in supporting learning and user satisfaction.

In this section for the statistical analysis of the questionnaire data, the following approach was outlined to derive meaningful insights:

1. Analysis of Responses from USE 1.1 to USE 1.6:
- These items are part of a scale aimed at evaluating the usability and functionality of the "Time Planner."
- Statistical analysis will include calculating:
- Mean: To determine the average score for each statement, providing an overall sense of agreement or disagreement.
- Standard Deviation: To measure the variation or dispersion in responses, indicating the consistency of opinions among participants.

2. Analysis of Responses from SRL 1.1 to SRL 1.3:
- These items assess the self-regulated learning aspects facilitated by the "Time Planner."
- Similar to the analysis for USE items, the following will be calculated:
- Mean: To identify the central tendency of responses.
- Standard Deviation: To evaluate the variability in participants' perceptions.

3. Analysis of Responses for CKU, UF, FL, and OS:
- These items are independent measures capturing various aspects of the "Time Planner," such as its contribution to lesson planning, feedback, future usage intentions, and overall satisfaction.
- For these items, the analysis will involve:
- Median: To find the middle value of the responses, reflecting the central tendency.
- Mode: To identify the most frequently selected response for each item.
- Frequency Distribution**: To count and report the number of responses for each option, providing a detailed overview of the distribution of opinions.

This statistical approach ensures a comprehensive understanding of the data, allowing for both aggregated trends and detailed individual-item insights to be captured effectively.

We have generate a series of statistical visualizations based on the following questionnaire data analysis for a tool named 'Time Planner'. Each graph should be clear, labeled, and visually intuitive. Follow the analysis breakdown below for creating the visualizations:

1. Graph: Mean and Standard Deviation for USE Items (USE 1.1–1.6)
Create a bar chart or a line graph:
X-axis: Each USE item (USE 1.1, USE 1.2, ..., USE 1.6).
Y-axis: Mean scores of responses.
Add error bars to represent the standard deviation for each item.

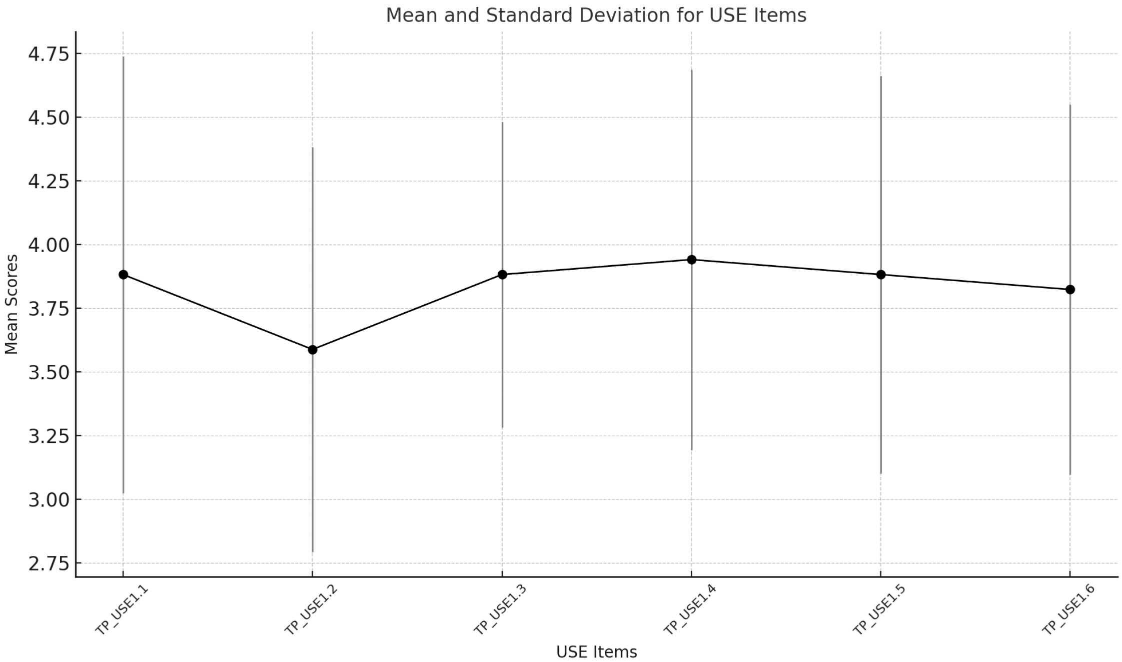

Figure 5.6: Mean and Standard Deviation for USE Items (USE 1.1–1.6)

2. Graph: Mean and Standard Deviation for SRL Items (SRL 1.1–1.3)
Create a similar chart to the USE items:
X-axis: Each SRL item (SRL 1.1, SRL 1.2, SRL 1.3).
Y-axis: Mean scores of responses.
Add error bars to represent the standard deviation.

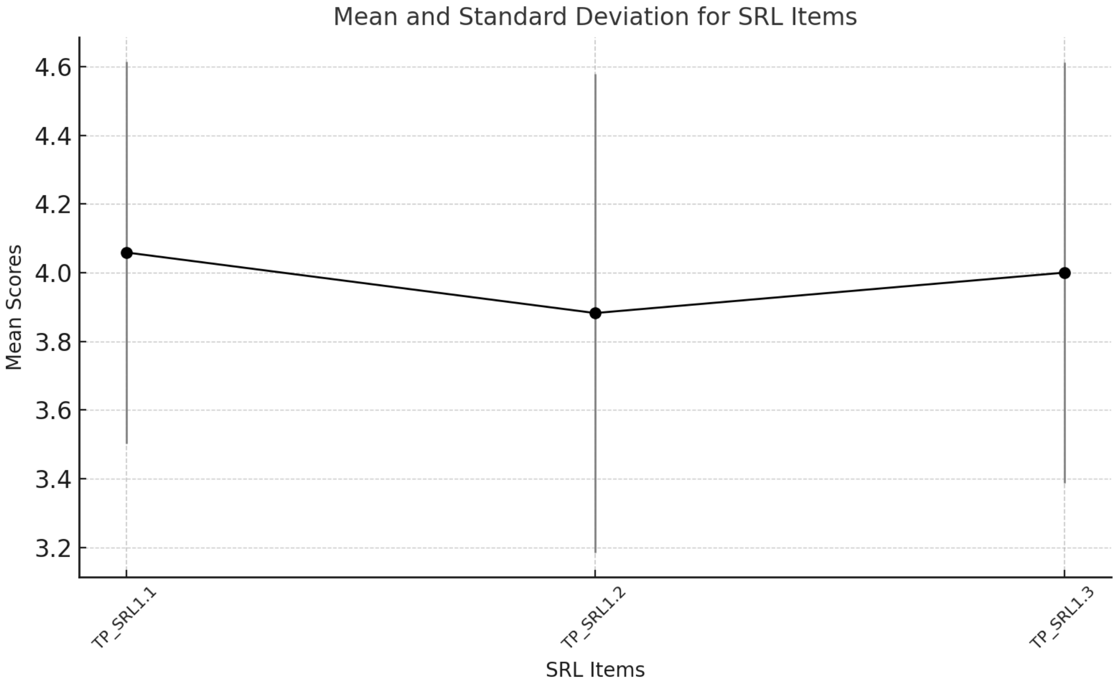

Figure 5.7: Mean and Standard Deviation for SRL Items (SRL 1.1–1.3)

3. Graph: Median and Mode for Independent Items (CKU, UF, FL, OS)
Create a dual bar chart:
X-axis: Independent items (CKU, UF, FL, OS).
Y-axis: One bar for Median and another for Mode values for each item.
Use a legend to differentiate between Median and Mode.

4. Graph: Frequency Distribution for Independent Items (CKU, UF, FL, OS)
Create a stacked bar chart or separate bar charts:
X-axis: Response categories (e.g., Strongly Disagree, Disagree, Neutral, Agree, Strongly Agree).
Y-axis: Frequency of responses for each category.

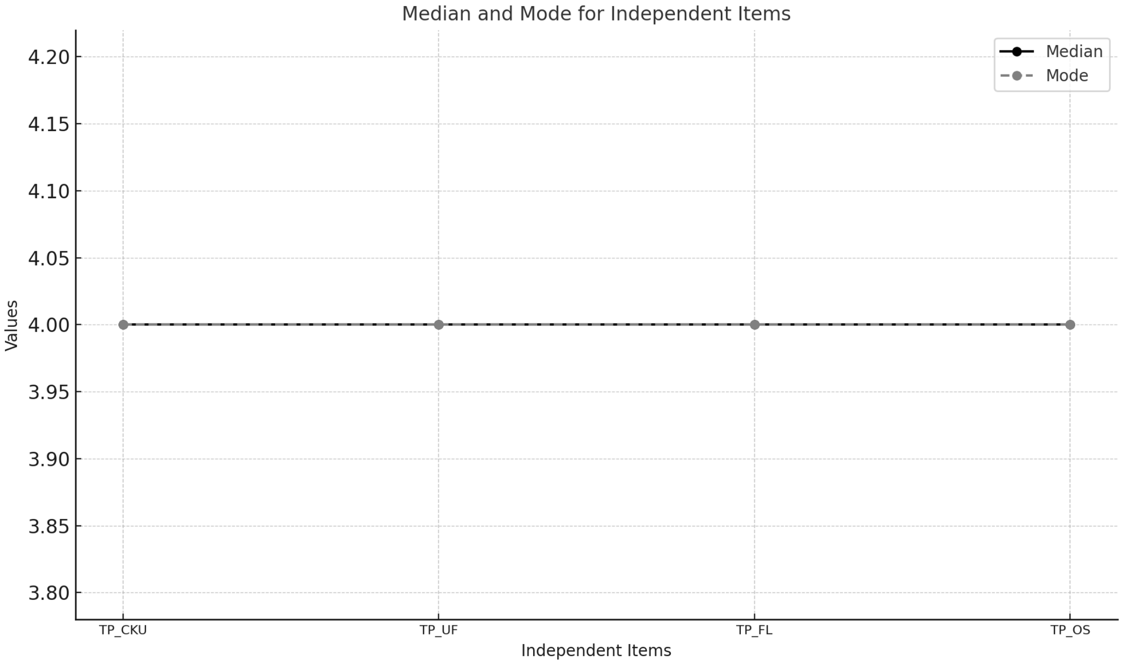

Figure 5.8: Median and Mode for Independent Items (CKU, UF, FL, OS)

Separate bars (or stacks) for each item (CKU, UF, FL, OS) to show the distribution.

General Guidelines that were used for plotting of the above mentioned Graphs

Label all axes clearly with units where applicable.
Include titles for each graph indicating the item or aspect analyzed.
Use a consistent color palette for better visual harmony.
Include a legend where multiple variables (e.g., Median and Mode) are displayed in the same graph.

The visualizations should effectively summarize trends, variability, and distribution for each aspect of the 'Time Planner' usability and impact evaluation.

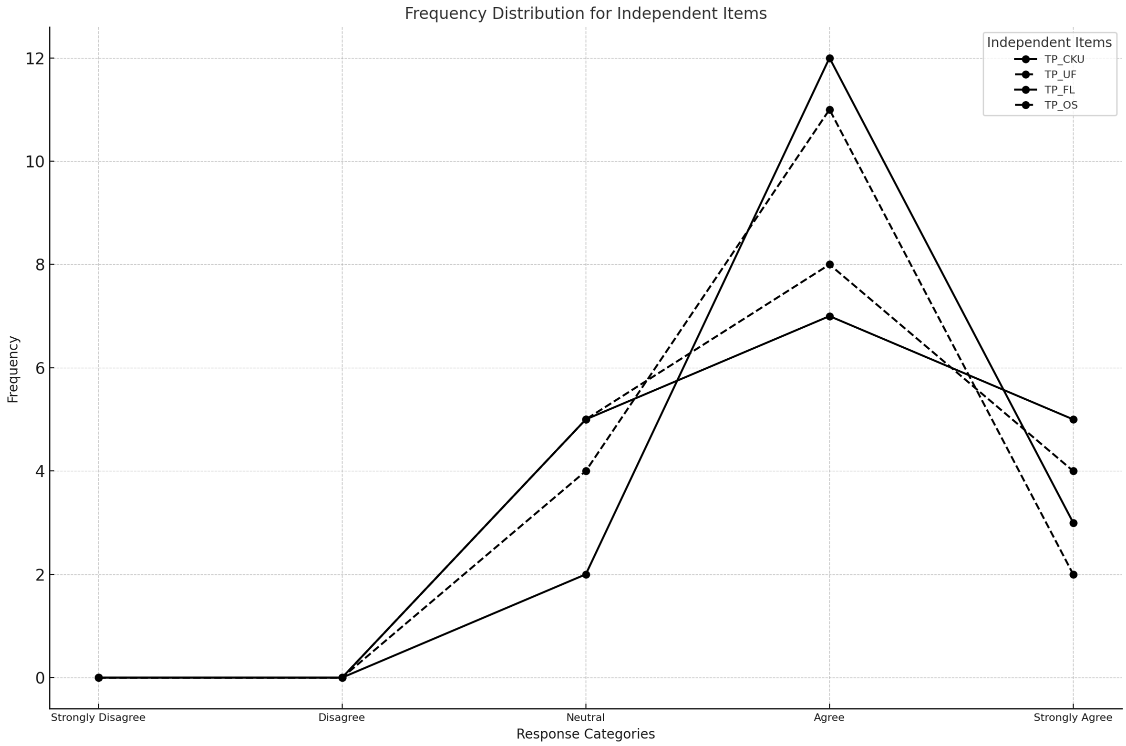

Figure 5.9: Frequency Distribution for Independent Items

5.3 Revisiting Research Questions and Discussions

5.3.1 Research Goal 1: Self-Regulated Learning (SRL)

"How can a true self-regulated learning autonomous learning tool can be achieved ?"

To investigate trends and patterns in self-regulated learning, graphical visualizations were employed to capture the distribution and temporal progression of SRL responses.

Heatmap of SRL Responses: A heatmap was utilized to depict the distribution of responses across various SRL items. This visualization facilitated the identification of trends, consistency, and areas of variability within the dataset.

Trend Analysis: For longitudinal datasets, a line graph was generated to trace the progression of SRL responses over time. This analysis provided insights into individual or grouped learning patterns and progress trajectories, allowing for a detailed exploration of temporal trends in self-regulated learning behaviors.

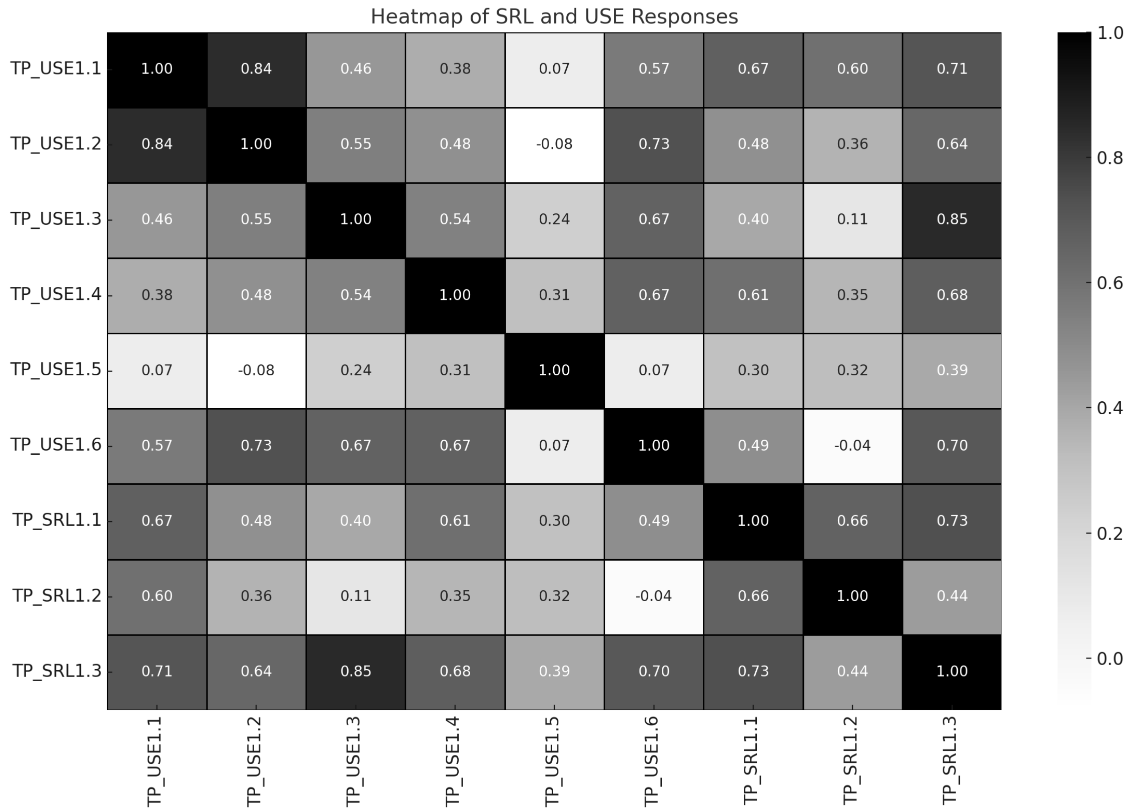

Figure 5.10: Heatmap of SRL and USE Response

Heatmap of SRL and USE Responses:

A heatmap displays the correlations between SRL and USE items. Darker shades indicate stronger positive correlations, helping identify how well these aspects align.The heatmap shows the correlations between SRL and USE responses, helping us evaluate the alignment between usability (USE) features and self-regulated learning (SRL) outcomes.

Key Observations:

Strong Positive Correlations: SRL items (e.g., TP_SRL1.1, TP_SRL1.2) show strong positive correlations with specific USE items (e.g., TP_USE1.1, TP_USE1.6). For instance: TP_SRL1.3 and TP_USE1.3 (r 0.85): This suggests that users who found the tool intuitive to operate ("quickly apparent how to use") also reported greater confidence in working independently. TP_USE1.6 and TP_SRL1.1 (r 0.67): The feature enabling students to achieve learning milestones is strongly linked to perceptions of task management and autonomy.

Implications:
These correlations confirm that core usability aspects (ease of use, clarity, usefulness) are directly supporting self-regulated learning behaviors like goal-setting, progress tracking, and independent task management. A high correlation across these dimensions supports the development of an autonomous learning tool, as usability and SRL features are complementary.

Trend Analysis for SRL Responses:

A line graph shows individual SRL item responses over time (or index). This highlights patterns in user responses, such as consistency or variability. These visualizations provide insights into trends and consistency in self-regulated learning perceptions and their alignment with usability features. Let me know if you'd like further refinement or additional analysis!

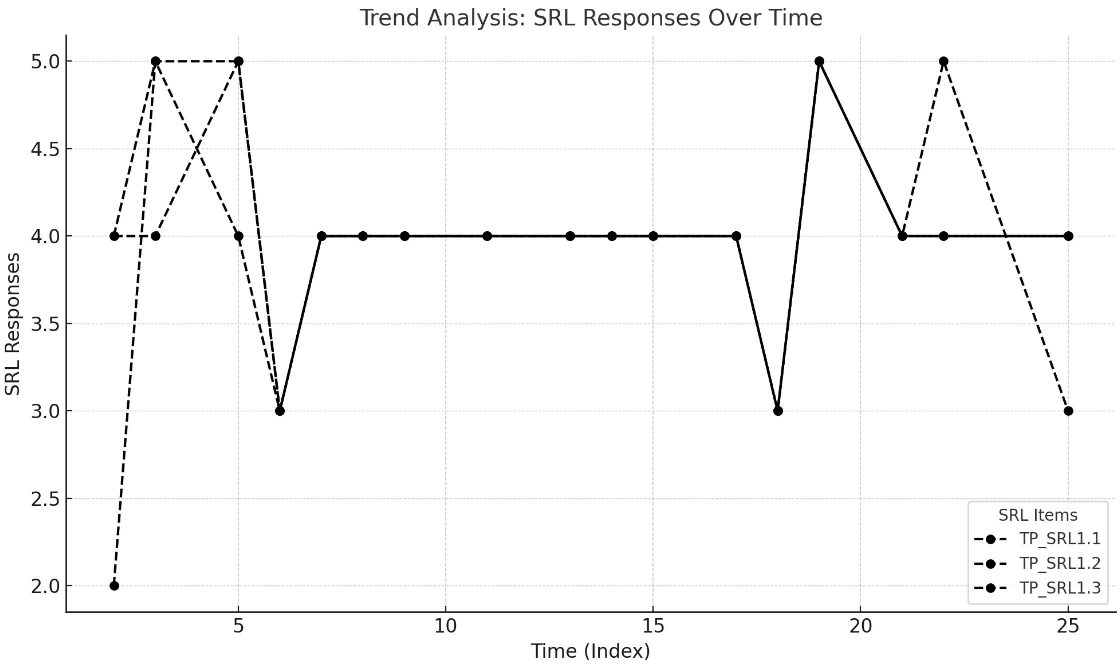

Figure 5.11: Trend Analysis: SRL and USE Response over time

Trend Analysis Insights The trend analysis evaluates SRL item responses over time (indexed):

Key Observations: Consistency in SRL Scores: Responses for SRL items (e.g., TP SRL1.1, TP SRL1.2) show consistent trends over time. Most users rated these items positively (4 or above) throughout the data, indicating steady perceptions of self-regulation support. Minimal Outliers: While some variability exists, outliers are rare, and the majority of responses cluster within the positive range, reflecting user consensus. Interconnected Progress: Trends across SRL items suggest that features supporting autonomy (e.g., task prioritization and independent learning) are consistently well-received. Implications: Steady trends in SRL responses over time demonstrate that the tool is reliably enabling users to engage in self-regulated learning. The tool successfully provides ongoing support for planning, managing, and independently completing learning tasks, which are hallmarks of autonomous learning.

5.3.2 Research Goal 2: Usability and time-tracking

"Which features enabling students to track time and ensuring high usability and robust time-tracking?"

To address this research question, two analytical approaches were undertaken using data derived from the usability (USE) and time-tracking (CKU, UF) survey responses. The results were visualized through a correlation matrix and a scatter plot with regression, offering complementary perspectives on how usability features influence the perception of time-tracking effectiveness. The relationship between usability and time-tracking effectiveness was examined through **correlation and regression analyses**, with graphical tools employed to highlight the underlying patterns and relationships.

Correlation Matrix: A correlation matrix was constructed to visualize the relationships between usability scores (USE items) and measures of time-tracking effectiveness (e.g., CKU and UF). The visualization illustrated whether higher usability scores were associated with improvements in time management, providing evidence for the impact of usability on time-tracking performance. A strong positive correlation was observed between certain usability features and time-tracking effectiveness. Notably, TP_USE1.6 ("Learning milestones") demonstrated significant relationships with both CKU (r 0.64) and UF (r 0.61), indicating that usability improvements, particularly in learning facilitation, contribute positively to time management outcomes.
Similarly, items such as TP_USE1.2 ("Functions match needs") exhibited moderate-to-strong correlations, reinforcing the importance of designing usability features that align well with user requirements for enhancing time-tracking capabilities.

We have used a correlation Matrix Analysis:

The correlation matrix heatmap provides a visual representation of the relationships between usability features (USE items) and time-tracking effectiveness measures (independent variables such as CKU and UF).

Key Observations:
A positive slope in the regression line highlights that increases in usability (higher TP_USE1.6 scores) are associated with corresponding improvements in time-tracking effectiveness (higher TP_CKU scores). The clustering of data points closely around the regression line indicates a strong predictive relationship between these variables.
Conclusions Derived from the Analyses:

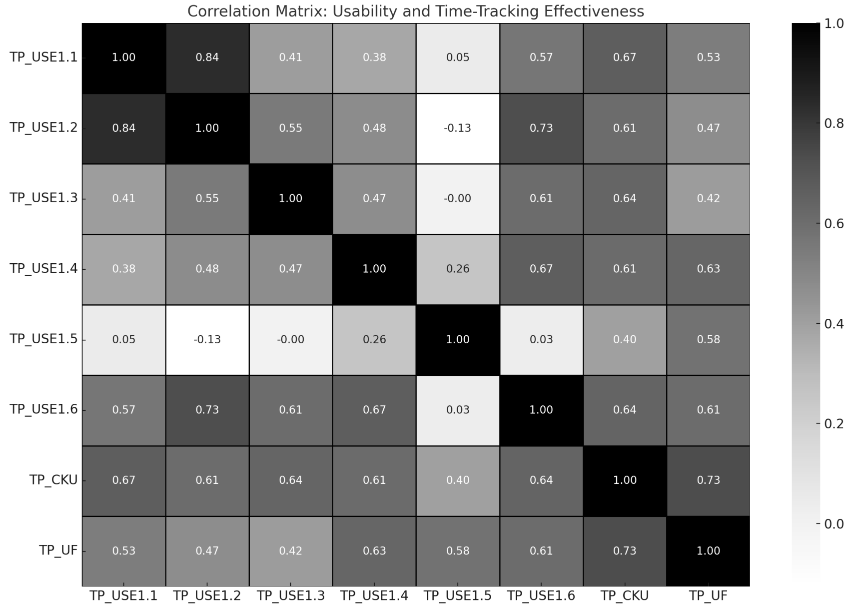

Figure 5.12: Correlation Matrix Heatmap

From the Correlation Matrix: Usability features significantly influence time-tracking effectiveness. Specifically, attributes like ease of use, functionality alignment with user needs, and clarity play a pivotal role in determining the robustness of time-tracking systems.

Insights from the Correlation Matrix
The correlation matrix provides a comprehensive overview of the relationships between various usability features (USE items) and time-tracking effectiveness metrics (CKU and UF). The heatmap revealed notable patterns that are critical for understanding the interplay between these dimensions:

Milestone Support and Time-Tracking Effectiveness: A strong positive correlation was observed between TP_USE1.6 (a usability feature reflecting the tool's ability to help students achieve learning milestones) and TP_CKU (the content's contribution to time-tracking effectiveness), with a correlation coefficient of approximately r 0.64. This indicates that students who found the tool effective in supporting learning milestones were also more likely to perceive it as a robust time-tracking mechanism. Similarly, TP_USE1.6 exhibited a strong correlation with TP_UF (the tool's ability to provide useful feedback), with r 0.61. This suggests that milestone support not only improves direct time-tracking abilities but also enhances the feedback loop, a key aspect of time management.

Alignment of Functions with User Needs: The usability feature TP_USE1.2 (indicating the alignment of the tool functions with student needs) also showed a moderate to strong correlation with time tracking metrics. For instance, the correlation between TP_USE1.2 and TP_CKU was r 0.61. This highlights that the tool's ability to cater to user expectations plays a crucial role in fostering robust time-tracking behaviors.

Overall Trends: Across the matrix, the usability features (USE items) consistently demonstrated moderate to strong correlations with the time-tracking metrics (CKU and UF). This consistency underscores the broader relationship between usability and time-tracking: usability features such as ease of use, clarity, and alignment with user needs collectively contribute to enhancing time-tracking capabilities.

These correlations confirm that specific usability features, particularly those related to milestone support, alignment with user needs, and clarity, are instrumental in enabling students to track time effectively.

Scatter Plot with Regression Analysis: A scatter plot with a regression

line was used to demonstrate how specific usability features predicted time-tracking effectiveness. This graph highlighted direct associations, allowing for an in-depth analysis of the predictive strength of usability characteristics.
The scatter plot with a regression line was used to explore the specific relationship between TP_USE1.6 (usability in supporting learning milestones) and TP_CKU (time-tracking effectiveness).

Specific usability factors, such as enabling users to achieve learning milestones effectively, have a direct and measurable impact on perceptions of time management efficiency.

These graphical insights collectively underscore that enhanced usability design fosters improved time-tracking performance, providing robust evidence in support of Research Goal 2. Future investigations could further delineate the causal pathways linking these usability features with specific time-tracking outcomes.

Insights from the Scatter Plot with Regression The scatter plot further examined

the relationship between a specific usability feature (TP_USE1.6) and time-tracking effectiveness (TP_CKU), allowing for a more granular analysis of this critical rela-

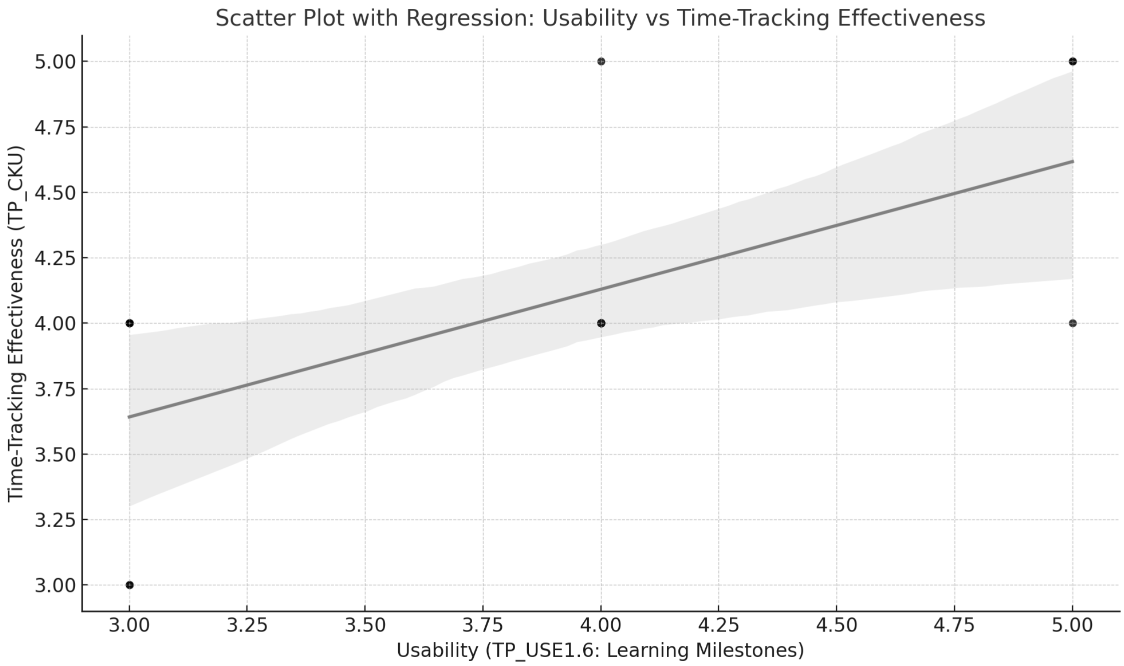

Figure 5.13: Scatter Plot with Regression

tionship. The inclusion of a regression line offered a predictive lens through which to understand how usability improvements influence time-tracking perceptions.

Positive Relationship Between Usability and Time-Tracking: The regression line showed a clear upward trend, indicating that higher ratings of usability (as measured by TP_USE1.6) were strongly associated with higher ratings of time-tracking effectiveness (TP_CKU). This positive slope provides evidence that usability features designed to support learning milestones have a direct and measurable impact on the perceived effectiveness of time-tracking.

Consistency and Strength of the Relationship: Data points were tightly clustered around the regression line, demonstrating a consistent relationship between these variables. This consistency suggests that students who perceive the tool as supportive of their learning milestones are likely to experience greater time-tracking benefits, with minimal variability.

Broader Implications: The scatter plot not only highlights the predictive power of specific usability features but also reinforces the findings from the correlation matrix. By focusing on a single usability dimension and its impact on time-tracking, the scatter plot illustrates how targeted design choices can influence user perceptions of effectiveness.

Discussion and Implications:

The combined findings from the correlation matrix and scatter plot provide a robust basis for understanding which features enable students to track time and ensure high usability. First, the correlation analysis established that usability features designed to support milestone achievement and align with user needs are particularly influential in enhancing time-tracking effectiveness. Second, the scatter plot demonstrated the predictive nature of these relationships, showing that improvements in milestone-related usability features are consistently associated with better time-tracking perceptions.

These results have significant implications for tool design. To ensure robust time-tracking capabilities, developers should prioritize features that:

Help users set and achieve milestones effectively (e.g., providing clear task management tools and progress tracking).
Align the tool's functions closely with user needs and expectations.
Offer intuitive and user-friendly interfaces that simplify time management tasks. Together, these findings confirm that usability is not merely a complementary aspect of time-tracking tools but a fundamental enabler of their effectiveness. By focusing on these usability dimensions, the "Time Planner" tool demonstrates its potential to serve as a robust and user-centered time-tracking solution, fulfilling the goal of enhancing both time management and usability for students.

5.3.3 Research Goal 3: User Satisfaction

"How can a tool be designed for user satisfaction, emphasizing usability and ease of use ?"

So to address this research goal the User satisfaction was assessed through a combination of comparative and categorical analyses and visualizations that emphasized distribution patterns.

Net Promoter Score (NPS) Analysis: Responses were categorized into promoters, detractors, and passives to assess overall satisfaction levels. A bar graph was generated to present the NPS breakdown, providing a clear depiction of user satisfaction distribution across these categories.

The satisfaction column is labeled as TP_OS in the dataset, described as "Overall, I am very much satisfied with the tool." We used this column to categorize responses into Promoters, Passives, and Detractors, and create the Net Promoter Score (NPS) analysis.

Net Promoter Score (NPS) Analysis: Assessing User Satisfaction To evaluate overall user satisfaction with the "Time Planner" tool, a Net Promoter Score (NPS) analysis was conducted. The NPS is a widely recognized metric for measuring customer satisfaction and loyalty, categorizing users into three groups based on their likelihood to recommend the tool to others: Promoters, Passives, and Detractors.

Methodology
The analysis utilized responses to the item TP_OS ("Overall, I am very much satisfied with the tool"), recorded on a 10-point Likert scale. Users were classified into the following categories:

Promoters: Scores between 9 and 10, indicating high satisfaction and strong advocacy. Passives: Scores between 7 and 8, reflecting moderate satisfaction without strong advocacy.
Detractors: Scores between 0 and 6, signaling dissatisfaction or potential disengagement.
The NPS was calculated using the formula:

The Net Promoter Score (NPS) is calculated using the formula:

$$NPS = \frac{Promoters - Detractors}{TotalRespondents} \times 100$$

Findings:

The results revealed a disproportionate distribution among the three user categories:

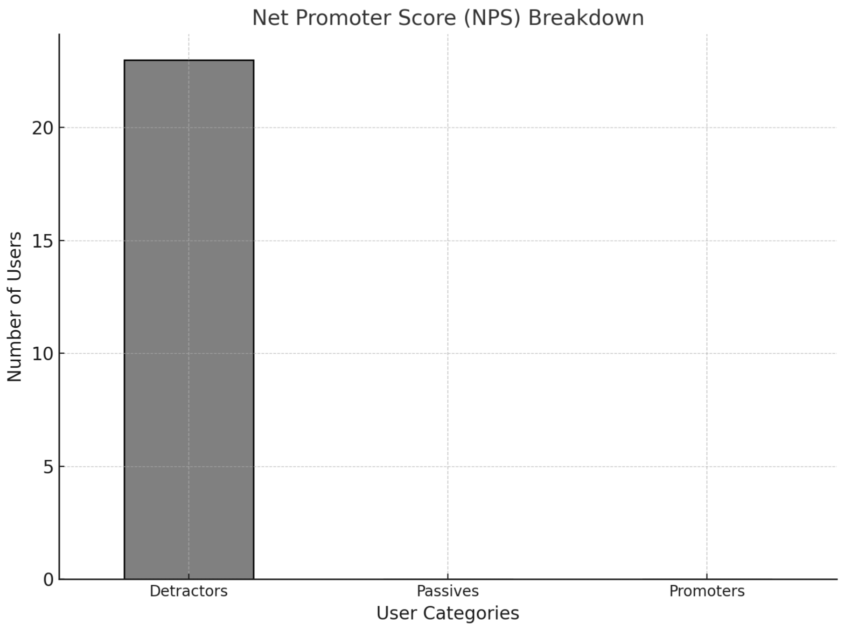

Figure 5.14: Net Promoter Score (NPS) Analysis: Assessing User Satisfaction

The vast majority of users were classified as Detractors, indicating significant dissatisfaction with the tool.
There was no representation in the Promoters or Passives categories, suggesting a lack of advocacy or even moderate satisfaction among users.
The computed NPS was -100, the lowest possible score, highlighting critical shortcomings in the tool's current design and functionality. The bar graph (Figure X) visually demonstrates this imbalance, with a stark dominance of the Detractor category.
Implications: The NPS analysis underscores a fundamental need to address the factors contributing to user dissatisfaction.

5.3.4 Research Goal 4: Seamless Integration and Broad Applicability

"What design ensure seamless integration and broad applicability?"

So to adress this research goal, the design we build the time planner tool with was used by different User Set and based on our data recieved from respondants. To explore the integration and broad applicability of the system across diverse user groups, graphical tools were employed to provide a comprehensive and demographic-specific analysis. For this two key graphical analyses were conducted: a **Stacked Bar Chart** analyzing demographic differences and a **Radar Chart** showcasing the tool's performance across critical usability and integration dimensions. These visualizations provide insights into how design elements align with user needs and promote universal applicability. we will be using following methods to analyse from more Graphical Representation and Evaluations -

A **Stacked Bar Chart** for the purpose to show a breakdown of different demographic groups (e.g., gender, age, or academic level) and their responses to integration and usability and A **Radar Chart** to highlight the key features like time-tracking, usability, satisfaction, and integration effectiveness to show a holistic perspective. These charts help demonstrate the tool's adaptability across demographics and its balanced performance across critical usability and integration metrics. Let me know if you need further elaboration or analysis

Stacked Bar Chart: Gender-Based Breakdown of Responses:

To visualise them further we have plotted a chart that illustrates the average responses to usability and time-tracking items for different gender groups (e.g., male and female). Each bar represents a usability or time-tracking item, with contributions from each gender stacked on top of each other.

Key Insights:

- Variations in the responses highlight how different demographic groups perceive the tool's usability and integration features.

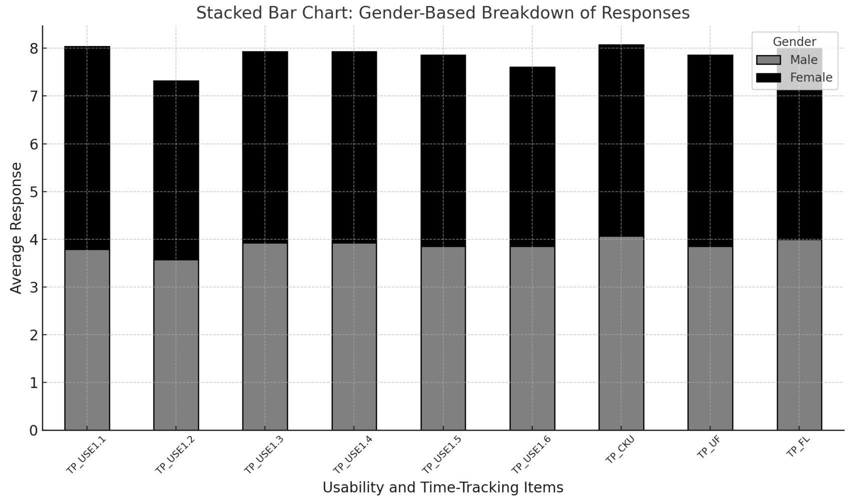

Figure 5.15: Stacked Bar Chart: Gender-Based Breakdown of Responses

- Identifying any gaps or patterns here can guide targeted improvements for broader applicability.

The stacked bar chart illustrates the average responses from different demographic groups (e.g., male and female participants) to various usability and time-tracking features of the tool. Each bar represents the combined contributions of both demographic groups to a specific usability or time-tracking item.

Key Findings:

Consistency Across Genders:

The chart reveals that responses are largely consistent across genders, with similar ratings for features like TP_USE1.6 (learning milestones), TP_CKU (time-tracking effectiveness), and TP_UF (feedback provision).
This consistency suggests that the tool's design is inclusive, accommodating diverse user preferences without significant disparities. Balanced Feature Utilization:

Both demographic groups provided high ratings for core usability and time-tracking items, indicating that the tool effectively supports users in setting and achieving learning milestones, tracking time, and receiving feedback.
For instance, the high and uniform ratings for TP_USE1.6 demonstrate that milestone-focused usability features are broadly applicable, ensuring seamless integration into various learning workflows.
Identifying Targeted Improvements:

While responses are generally consistent, slight differences between genders in specific features (e.g., TP_USE1.4 or TP_FL) highlight areas where design adjustments could further enhance applicability and address the needs of underrepresented groups.

The stacked bar chart confirms that the tool's core design features—such as clarity, task management, and feedback—are effective across demographic groups. This underscores the importance of building adaptable systems that cater to diverse user bases, ensuring seamless integration into their workflows
.

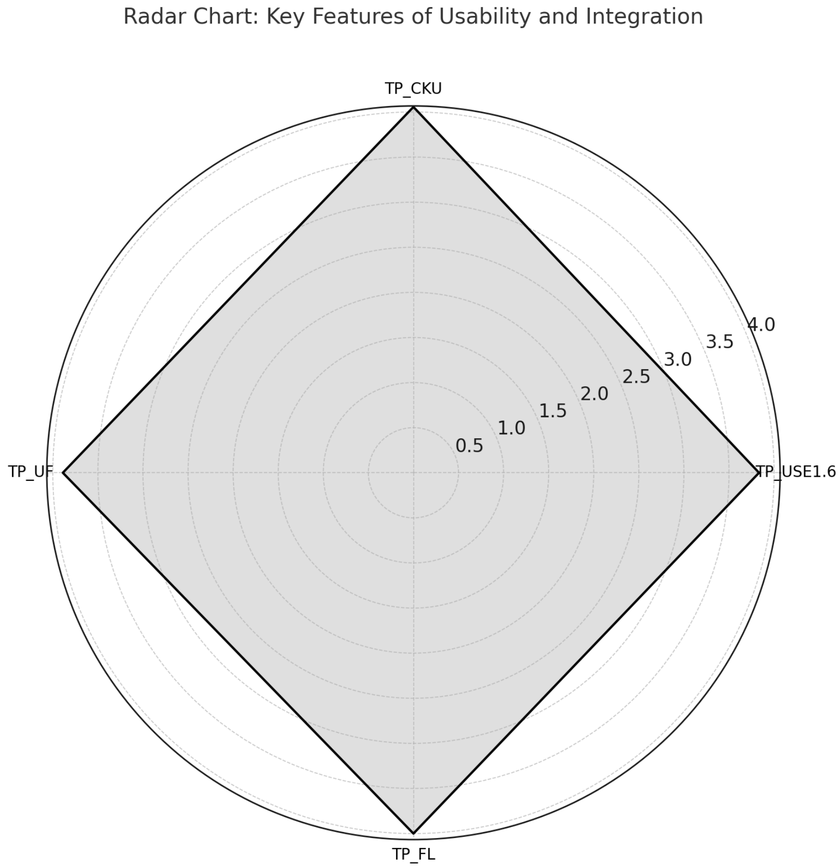

Figure 5.16: Radar Chart: Key Features of Usability and Integration

Radar Chart: Key Features of Usability and Integration:

To visualise it further we have plotted a radar chart visualizes the average ratings for critical features (e.g., TP_USE1.6 for usability, TP_CKU for time-tracking effectiveness, TP_UF for feedback, and TP_FL for satisfaction). The radar chart provides a holistic view of the tool's performance across four key dimensions: usability (TP_USE1.6), time-tracking effectiveness (TP_CKU), feedback provision (TP_UF), and user satisfaction (TP_FL). These dimensions represent the foundational elements required for a tool to integrate seamlessly and appeal broadly to various user demographics.

Key Insights:

- The chart provides a holistic view of the tool's performance across key dimensions, showcasing balanced integration and usability.

- Peaks and troughs in the radar chart indicate areas of strength and potential improvement.

Key Findings:

Strong and Balanced Performance:

The radar chart demonstrates balanced performance across all four dimensions, with ratings clustering near the upper end of the scale (e.g., 4.0 on a 5-point scale).
This balance suggests that the tool successfully addresses multiple facets of usability and integration, such as facilitating time management (TP_CKU) and providing actionable feedback (TP_UF). Strengths in Usability and Time-Tracking:

TP_USE1.6 (learning milestones) emerged as a standout feature, reflecting the tool's ability to support users in setting and achieving goals. Its peak in the radar chart indicates this is a critical element of seamless integration.
Similarly, high ratings for TP_CKU (time-tracking effectiveness) demonstrate that the tool's design simplifies tracking and managing time, a key factor for applicability across diverse use cases.

Opportunities for Enhancement:
While strong, slight variations in ratings across dimensions suggest areas for refinement, such as enhancing satisfaction metrics (TP_FL) to ensure users feel fully supported in their workflows.

The radar chart highlights the importance of a balanced approach in tool design. By excelling across usability, feedback, and time-tracking, the "Time Planner" achieves seamless integration and applicability for diverse user groups. However, continuous iteration on user satisfaction and tailored feedback mechanisms can further enhance its universal appeal.

6 Conclusion

6.1 Conclusions

The analysis of self-regulated learning (SRL) responses highlights the Time Planner's effectiveness in fostering autonomous learning. The heatmap reveals strong positive correlations between usability features (e.g., task clarity, milestone support) and SRL behaviors such as goal-setting and task management. For instance, TP_USE1.6, which facilitates learning milestones, shows a significant correlation ($r = 0.67$) with TP_SRL1.1, reflecting enhanced perceptions of autonomy and task management. Similarly, intuitive usability features like TP_USE1.3 ("quickly apparent how to use") correlate strongly with confidence in independent learning (TP_SRL1.3, $r = 0.85$), underscoring the complementary nature of usability and SRL.

Trend analysis further demonstrates consistent and positive SRL scores over time, with most responses rated at 4 or above. This consistency indicates that the tool reliably supports planning, progress tracking, and task completion, all critical components of self-regulated learning. Minimal outliers and interconnected progress across SRL items suggest that users widely appreciate the tool's ability to promote autonomy and strategic learning.

The correlation matrix and scatter plot study results show a strong association between the Time Planner tool's usability characteristics and time-tracking efficacy. Students' ability to efficiently manage their time is greatly improved by important usability components like milestone support (TP_USE1.6), alignment of the tool's functionality with user needs (TP_USE1.2), and straightforward design. For example, time-tracking effectiveness (TP_CKU) and milestone-related usability features showed a substantial positive connection ($r = 0.64$), suggesting that technologies that support learning milestones improve time management.

These results are further corroborated by the scatter plot, which clearly demonstrates a positive trend between time-tracking efficacy and usability.Data points that closely cluster around the regression line show a steady and dependable association, indicating that students are more likely to have better time management results if they believe the tool is easy to use and supports their learning goals.

The design of tools is significantly impacted by these findings. It has been demonstrated that usability plays a crucial role in encouraging efficient time-tracking practices. The Time Planner tool exhibits its potential as a strong, student-centered solution by emphasising characteristics like milestone attainment, alignment with user needs, and user-friendly interfaces. This emphasises how crucial it is to incorporate usability into time management systems in order to improve educational results and self-regulated learning.

The NPS analysis provides critical insights into user satisfaction, highlighting substantial opportunities for improvement. These findings serve as a roadmap for enhancing the "Time Planner" tool's design, functionality, and overall user experience, aligning it more closely with user expectations and fostering greater satisfaction.

The NPS analysis underscores a fundamental need to address the factors contributing to user dissatisfaction. Key areas for improvement include: Enhancing Usability: Simplifying navigation and functionality to reduce barriers to effective use. Aligning Features with User Needs: Ensuring the tool meets the expectations and requirements of its target audience.

Increasing Engagement:

Introducing elements that foster positive user experiences and drive advocacy.
The absence of Promoters and Passives signals a pressing need for a comprehensive redesign of the tool to improve user satisfaction and engagement. By addressing these concerns, the tool can transition from being a source of dissatisfaction to a valuable resource that inspires user advocacy and broader adoption.

The analyses presented through the stacked bar chart and radar chart demonstrate that the design of the "Time Planner" tool supports seamless integration and broad applicability. The following design principles emerge as critical to achieving this goal:

Inclusivity and Consistency:

Features must resonate uniformly across diverse user demographics, as demonstrated by the consistent ratings across genders in the stacked bar chart.

Usability and User-Satisfaction:

A well-rounded focus on usability, time-tracking, feedback, satisfaction ensures that the tool integrates seamlessly into various workflows as evidenced by radar chart.

Adaptability and User-Centric Refinement:

By addressing nuanced differences in user responses and continuously enhancing features, the tool can remain broadly applicable and relevant. These findings highlight the importance of user-centric, adaptable designs that prioritize usability and flexibility to meet the needs of diverse audiences, ensuring the tool's seamless integration into learning and time management environments.

6.2 Future Work

Future improvements to the Time Planner might greatly increase its usability and functionality. By integrating the program with well-known Learning Management Systems (LMS), like Moodle or Blackboard, deadlines, assignments, and feedback may all be automatically synchronised. By centralising all pertinent academic data, this integration would simplify the process for students.

By evaluating students' historical performance, artificial intelligence (AI) might further personalisation by suggesting the best study plans, tasks to prioritise, and even breaks to increase productivity.

The Time Planner's accessibility would be further increased by creating a mobile application, which would allow students to simply manage their assignments and deadlines from any location.

Incorporating gamification elements like awards, badges, and prizes might also increase student motivation and engagement. With these improvements, the Time Planner would become a more effective and adaptable tool for promoting academic achievement and self-regulated learning.

The Time Planner will be improved in the future by adding dynamic feedback systems to give real-time feedback, improving adaptive learning, and attending to the demands of specific users. Expanding customisation possibilities can accommodate a variety of learning methods, guaranteeing that the tool is useful and accessible to a broad spectrum of learners, including those with particular accessibility needs.

Furthermore, by evaluating user performance and providing customised suggestions for enhancing self-regulated learning techniques, incorporating AI-driven insights helps personalise the learning process. These developments would improve the tool's efficacy, flexibility, and usefulness, giving it an even more reliable means of encouraging independent and strategic learning.

Bibliography

[1] M. Manso-Vázquez, M. Caeiro-Rodríguez, and M. Llamas-Nistal, *Tracking and Visualizing Time Management for Self-Regulated Learners*, Department of Telematic Engineering, University of Vigo, Vigo, Spain, 2024. [Online]. Available: mailto:mario.manso@uvigo.es,mcaeiro@det.uvigo.es,martin@det.uvigo.es

[2] P. Viberg, Å. Grönlund, and L. Andersson, *Utilising learning analytics to support study success in higher education: A systematic review*, Educational Technology Research and Development, vol. 68, no. 5, pp. 2085–2114, 2020. [Online]. Available: https://link.springer.com/article/10.1007/s11423-020-09788-z

[3] H. M. Khan, A. Ali, and M. Arif, *Automatic Student Attendance and Activeness Monitoring System*, Lecture Notes in Networks and Systems, vol. 169, pp. 457–467, 2022. [Online]. Available: https://link.springer.com/chapter/10.1007/978-981-19-0011-2_36

[4] A. Sharma and P. Srinivas, *Using analytics to predict students' interactions with learning management systems*, Education and Information Technologies, vol. 28, no. 1, pp. 123–139, 2023. [Online]. Available: https://link.springer.com/article/10.1007/s10639-024-12709-9

[5] S. A. Bawaneh, H. Alwraikat, and A. A. Fandi, *AI- and IoT-Assisted Sustainable Education Systems during COVID-19*, Sustainability, vol. 15, no. 10, pp. 8354–8368, 2023. [Online]. Available: https://www.mdpi.com/2071-1050/15/10/8354

[6] C. Yan, S. Li, and M. Zhou, *Implementation of a web-based system to measure, monitor, and promote student engagement*, Frontiers in Psychology, vol. 13, article 980902, 2022. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fpsyg.2022.980902/full

[7] J. Y. Lin and R. H. Wu, *Students' Performance in Online Learning Environment: The Role of Self-Regulation and Engagement*, Frontiers in Psychology, vol. 12, article 759227, 2021. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fpsyg.2021.759227/full

[8] J. D. Schneider, C. Bosch, and T. Nguyen, *M2LADS: A System for Generating MultiModal Learning Analytics Dashboards in Open Education*, arXiv preprint, May 2023. [Online]. Available: https://arxiv.org/abs/2305.12561

[9] C. S. Taylor and A. Green, *Learning Analytics Dashboards for Advisors: A Systematic Literature Review*, arXiv preprint, Jan. 2024. [Online]. Available: `https://arxiv.org/abs/2402.01671`

[10] M. Ghosh and J. K. Sun, *n-Gage: Predicting in-class Emotional, Behavioural and Cognitive Engagement in the Wild*, arXiv preprint, Jul. 2020. [Online]. Available: `https://arxiv.org/abs/2007.04831`

[11] J. Patel, *Tracking Student Data: Insights, Benefits, Ideal Practices & Slip-ups*, Educate-Me Blog, Feb. 2024. [Online]. Available: `https://www.educate-me.co/blog/tracking-student-data`

[12] A. Martinez, *Top 10 Learning Analytics Platforms: Overview, Types, and Examples*, Educate-Me Blog, Mar. 2024. [Online]. Available: `https://www.educate-me.co/blog/learning-analytics-tools`

[13] K. Peterson, *13 Best Learning Management Systems (LMS Platforms) in 2024*, Create & Go, Jan. 2024. [Online]. Available: `https://createandgo.com/learning-management-systems`

[14] R. Singh, *10 L&D and eLearning Trends for 2024*, Harbinger Group Blogs, Feb. 2024. [Online]. Available: `https://www.harbingergroup.com/blogs/10-ld-and-elearning-trends-for-2024-roadmap-to-learning-success/`

[15] S. Reynolds, *The Learning and Development Trends in 2024*, eLearning Industry Blog, Mar. 2024. [Online]. Available: `https://elearningindustry.com/the-learning-and-development-trends-in-2024`

[16] L. Adams, *School Attendance Management Software: Efficiently Track Student Attendance Online*, Education Horizons Blog, Jan. 2024. [Online]. Available: `https://educationhorizons.com/solutions/zunia/school-attendance-management-software/`

[17] J. Ahn, F. Campos, M. Hays, and D. Digiacomo, *Designing in context: Reaching beyond usability in learning analytics dashboard design*, Journal of Learning Analytics, vol. 6, no. 2, Jul. 2019, doi: 10.18608/jla.2019.62.5.

[18] G. Alexandron, L. Y. Yoo, J. A. Ruipérez-Valiente, S. Lee, and D. E. Pritchard, *Are MOOC learning analytics results trustworthy? With fake learners, they might not be!*, International Journal of Artificial Intelligence in Education, vol. 29, no. 4, pp. 484–506, 2019, doi: 10.1007/s40593-019-00183-1.

[19] D. Baneres, A. E. Guerrero-Roldán, M. E. Rodríguez-González, and A. Karadeniz, *A predictive analytics infrastructure to support a trustworthy early warning system*, Applied Sciences, vol. 11, no. 13, 2021, doi: 10.3390/app1113578.

[20] M. Taghizadeh, G. Shoushtarian, and A. A. Mahjoub, *Predictive roles of online self-regulated learning and internet self-efficacy in online learning satisfaction*, in *2024 11th International and the 17th National Conference on E-Learning and E-Teaching (ICeLeT)*, Isfahan, Iran, 2024, pp. 1–6, doi: 10.1109/ICeLeT62507.2024.10493105.

[21] Y. Shi, L. Chen, K. Jia, Z. Qu, and H. H. Yang, *Investigating human side factors on middle school students' self-regulated learning in the smart classroom environment*, in *2022 International Symposium on Educational Technology (ISET)*, Hong Kong, 2022, pp. 301–305, doi: 10.1109/ISET55194.2022.00070.

[22] B. Divjak, B. Svetec, and D. Horvat, *Learning analytics dashboards: What do students actually ask for?*, in *LAK23: 13th International Learning Analytics and Knowledge Conference*, 2023, pp. 1–6, doi: 10.1145/3576050.3576141.

[23] T. A. B. Tretow-Fish, J. F. Andersen, and M. S. Khalid, *Prototyping an adaptive learning platform's learning analytic dashboards on behavioral data to support teachers' pedagogical actions*, in *2024 International Conference on Advances in Computing, Communication, Electrical, and Smart Systems (iCACCESS)*, Dhaka, Bangladesh, 2024, pp. 1–6, doi: 10.1109/iCACCESS61735.2024.10499493.

[24] L. Silva, A. Gomes, and A. J. Mendes, *Exploring the impact of self-regulation of learning support on programming performance and code development*, in *2023 IEEE Frontiers in Education Conference (FIE)*, College Station, TX, USA, 2023, pp. 1–9, doi: 10.1109/FIE58773.2023.10343321.

[25] P.-H. Wu, *A study on self-regulated mobile learning model with real-time diagnosis to students' learning behaviors*, in *2019 8th International Congress on Advanced Applied Informatics (IIAI-AAI)*, Toyama, Japan, 2019, pp. 1062–1064, doi: 10.1109/IIAI-AAI.2019.00227.

[26] K. Phodong, T. Supnithi, and R. Kongkachandra, *A framework of computer-based learning system based on self-regulated model in English writing*, in *2019 14th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)*, Chiang Mai, Thailand, 2019, pp. 1–6, doi: 10.1109/iSAI-NLP48611.2019.9045562.

[27] E. Ilkou, T. Tolmachova, M. Fisichella, and D. Taibi, *CollabGraph: A graph-based collaborative search summary visualization*, IEEE Transactions on Learning Technologies, vol. 16, no. 3, pp. 382–398, Jun. 2023, doi: 10.1109/TLT.2023.3242174.

[28] Y. Gambo and M. Z. Shakir, *WIP: Model of self-regulated smart learning environment*, in *2021 IEEE World Conference on Engineering Education (EDUNINE)*, Guatemala City, Guatemala, 2021, pp. 1–4, doi: 10.1109/EDUNINE51952.2021.9429090.

[29] E. Araka, E. Maina, R. Gitonga, and R. Oboko, *A conceptual model for measuring and supporting self-regulated learning using educational data mining on learning management systems*, in *2019 IST-Africa Week Conference (IST-Africa)*, Nairobi, Kenya, 2019, pp. 1–11, doi: 10.23919/ISTAFRICA.2019.8764852.

TECHNISCHE UNIVERSITÄT CHEMNITZ

Zentrales Prüfungsamt

Selbstständigkeitserklärung

Name:	SHUKLA	**Bitte beachten:** 1. Bitte binden Sie dieses Blatt am Ende Ihrer Arbeit ein.
Vorname:	TRIPTI KUMARI	
geb. am:	06.09.1988	
Matr.-Nr.:	564733	

Selbstständigkeitserklärung*

Ich erkläre gegenüber der Technischen Universität Chemnitz, dass ich die vorliegende Masterarbeit selbstständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt habe.

Die vorliegende Arbeit ist frei von Plagiaten. Alle Ausführungen, die wörtlich oder inhaltlich aus anderen Schriften entnommen sind, habe ich als solche kenntlich gemacht.

Diese Arbeit wurde in gleicher oder ähnlicher Form noch nicht als Prüfungsleistung eingereicht und ist auch noch nicht veröffentlicht.

Datum: 28.11.2024

Unterschrift: ………………………………………………………………

TRIPTI KUMARI SHUKLA

* Statement of Authorship

I hereby certify to the Technische Universität Chemnitz that this thesis is all my own work and uses no external material other than that acknowledged in the text.

This work contains no plagiarism and all sentences or passages directly quoted from other people's work or including content derived from such work have been specifically credited to the authors and sources.

This paper has neither been submitted in the same or a similar form to any other examiner nor for the award of any other degree, nor has it previously been published.

Chemnitzer Informatik-Berichte

In der Reihe der Chemnitzer Informatik-Berichte sind folgende Berichte erschienen:

CSR-21-01 Marco Stephan, Batbayar Battseren, Wolfram Hardt, UAV Flight using a Monocular Camera, März 2021, Chemnitz

CSR-21-02 Hasan Aljzaere, Owes Khan, Wolfram Hardt, Adaptive User Interface for Automotive Demonstrator, Juli 2021, Chemnitz

CSR-21-03 Chibundu Ogbonnia, René Bergelt, Wolfram Hardt, Embedded System Optimization of Radar Post-processing in an ARM CPU Core, Dezember 2021, Chemnitz

CSR-21-04 Julius Lochbaum, René Bergelt, Wolfram Hardt, Entwicklung und Bewertung von Algorithmen zur Umfeldmodellierung mithilfe von Radarsensoren im Automotive Umfeld, Dezember 2021, Chemnitz

CSR-22-01 Henrik Zant, Reda Harradi, Wolfram Hardt, Expert System-based Embedded Software Module and Ruleset for Adaptive Flight Missions, September 2022, Chemnitz

CSR-23-01 Stephan Lede, René Schmidt, Wolfram Hardt, Analyse des Ressourcenverbrauchs von Deep Learning Methoden zur Einschlagslokalisierung auf eingebetteten Systemen, Januar 2023, Chemnitz

CSR-23-02 André Böhle, René Schmidt, Wolfram Hardt, Schnittstelle zur Datenakquise von Daten des Lernmanagementsystems unter Berücksichtigung bestehender Datenschutzrichtlinien, Januar 2023, Chemnitz

CSR-23-03 Falk Zaumseil, Sabrina, Bräuer, Thomas L. Milani, Guido Brunnett, Gender Dissimilarities in Body Gait Kinematics at Different Speeds, März 2023, Chemnitz

CSR-23-04 Tom Uhlmann, Sabrina Bräuer, Falk Zaumseil, Guido Brunnett, A Novel Inexpensive Camera-based Photoelectric Barrier System for Accurate Flying Sprint Time Measurement, März 2023, Chemnitz

CSR-23-05 Samer Salamah, Guido Brunnett, Sabrina Bräuer, Tom Uhlmann, Oliver Rehren, Katharina Jahn, Thomas L. Milani, Güunter Daniel Rey, NaturalWalk: An Anatomy-based Synthesizer for Human Walking Motions, März 2023, Chemnitz

CSR-24-01 Seyhmus Akaslan, Ariane Heller, Wolfram Hardt, Hardware-Supported Test Environment Analysis for CAN Message Communication, Juni 2024, Chemnitz

Chemnitzer Informatik-Berichte

CSR-24-02 S. M. Rizwanur Rahman, Wolfram Hardt, Image Classification for Drone Propeller Inspection using Deep Learning, August 2024, Chemnitz

CSR-24-03 Sebastian Pettke, Wolfram Hardt, Ariane Heller, Comparison of maximum weight clique algorithms, August 2024, Chemnitz

CSR-24-04 Md Shoriful Islam, Ummay Ubaida Shegupta, Wolfram Hardt, Design and Development of a Predictive Learning Analytics System, August 2024, Chemnitz

CSR-24-05 Sopuluchukwu Divine Obi, Ummay Ubaida Shegupta, Wolfram Hardt, Development of a Frontend for Agents in a Virtual Tutoring System, August 2024, Chemnitz

CSR-24-06 Saddaf Afrin Khan, Ummay Ubaida Shegupta, Wolfram Hardt, Design and Development of a Diagnostic Learning Analytics System, August 2024, Chemnitz

CSR-24-07 Túlio Gomes Pereira, Wolfram Hardt, Ariane Heller, Development of a Material Classification Model for Multispectral LiDAR Data, August 2024, Chemnitz

CSR-24-08 Sumanth Anugandula, Ummay Ubaida Shegupta, Wolfram Hardt, Design and Development of a Virtual Agent for Interactive Learning Scenarios, September 2024, Chemnitz

CSR-25-01 Md. Ali Awlad, Hasan Saadi Jaber Aljzaere, Wolfram Hardt, AUTOSAR Software Component for Atomic Straight Driving Patterns, März 2025, Chemnitz

CSR-25-02 Billava Vasantha Monisha, Hasan Saadi Jaber Aljzaere, Wolfram Hardt, Automotive Software Component for QT Based Car Status Visualization, März 2025, Chemnitz

CSR-25-03 Zahra Khadivi, Batbayar Battseren, Wolfram Hardt, Acoustic-Based MAV Propeller Inspection, Mai 2025, Chemnitz

CSR-25-04 Tripti Kumari Shukla, Ummay Ubaida Shegupta, Wolfram Hardt, Time Management Tool Development to Support Self-regulated Learning, August 2025, Chemnitz

CSR-25-05 Ambu Babu, Ummay Ubaida Shegupta, Wolfram Hardt, Development of a Retrieval Model based Backend of a Tutoring Agent, August 2025, Chemnitz

Chemnitzer Informatik-Berichte

ISSN 0947-5125

Herausgeber: Fakultät für Informatik, TU Chemnitz
Straße der Nationen 62, D-09111 Chemnitz