**Project Proposal: Timetable Management System - “Planify” by Red Mouse**

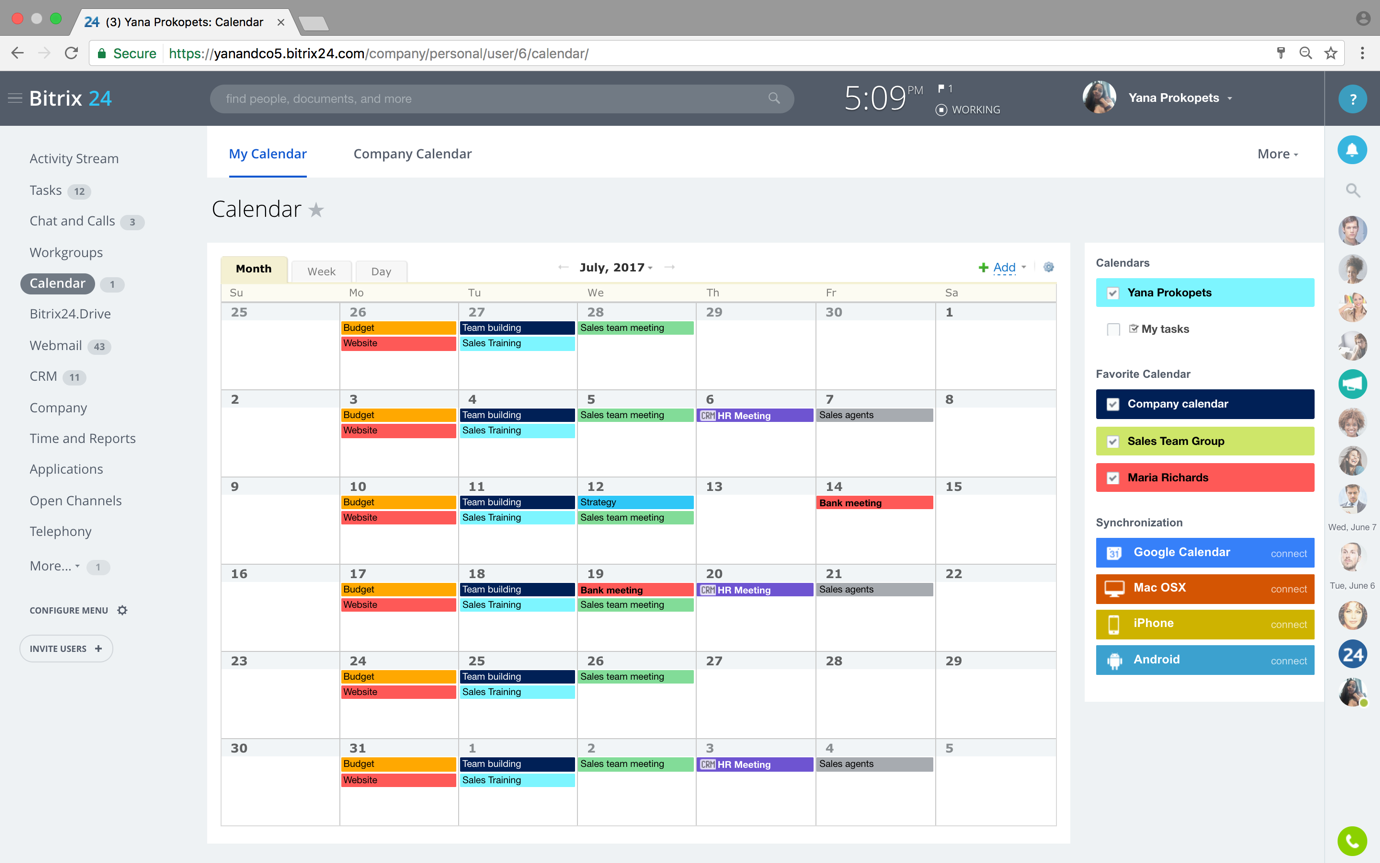
**Project Relevance**

Effective time management is essential in both professional and educational environments. With increasing complexity in scheduling, a reliable timetable management system can streamline operations, reduce conflicts, and improve productivity. This project addresses the challenges of manual scheduling and inefficient management by offering an intuitive digital solution. It is especially relevant for organizations, educational institutions, and teams aiming to optimize their workflows.

Competitors such as Notion, Bitrix, and WEEEK dominate the market, yet our proposed system introduces specialized features tailored to enhance usability and flexibility for users requiring robust timetable solutions. These include synchronized calendars, CRUD functionalities for users, tables, and teams, and customizable table templates, distinguishing our solution in the competitive landscape.

**Analysis of Competitors**

* **Notion:** Renowned for its versatility and collaborative features, Notion offers extensive customization but lacks a dedicated timetable management focus, which can be overwhelming for users seeking straightforward solutions.  
    
  A screenshot of a roadmap template

  Description automatically generated
* **Bitrix:** Provides comprehensive business management tools, including task management and calendars, but its complexity and steep learning curve deter smaller organizations or individuals.  
    
  
* **WEEEK:** Focused on task and project management, WEEEK offers a user-friendly interface but does not specialize in timetable-specific functionalities.  
    
  A screenshot of a computer

  Description automatically generated

**How Our System Is Better:**

* **Focus on Timetable Management:** Unlike competitors that offer general tools, our system is dedicated to solving timetable-specific challenges.
* **User-Friendly Customization:** Intuitive table templates make scheduling easier and faster, which is a unique offering compared to the complex interfaces of competitors.
* **Real-Time Synchronization:** Our synchronized calendar ensures updates are instantly reflected across all devices, offering a seamless user experience.
* **Simplified Team Collaboration:** CRUD operations for teams allow smooth collaboration, a feature often buried or underdeveloped in competitors’ systems.

**Project Audience**

The primary audience for this system includes:

* **Educational Institutions:** Schools, colleges, and universities requiring automated scheduling solutions for classes, exams, and events.
* **Organizations:** Teams and departments needing streamlined scheduling for meetings, projects, and resource allocation.
* **Individuals:** Professionals managing personal and professional schedules who value synchronization and customization.

**Project Features**

**Core Functionalities:**

* **CRUD Operations for Users:** Create, read, update, and delete user profiles, ensuring user management.
* **CRUD Operations for Tables:** Manage tables effectively, allowing for flexible scheduling options.
* **CRUD Operations for Teams:** Add and manage teams seamlessly, catering to collaborative environments.

**Supplementary Features:**

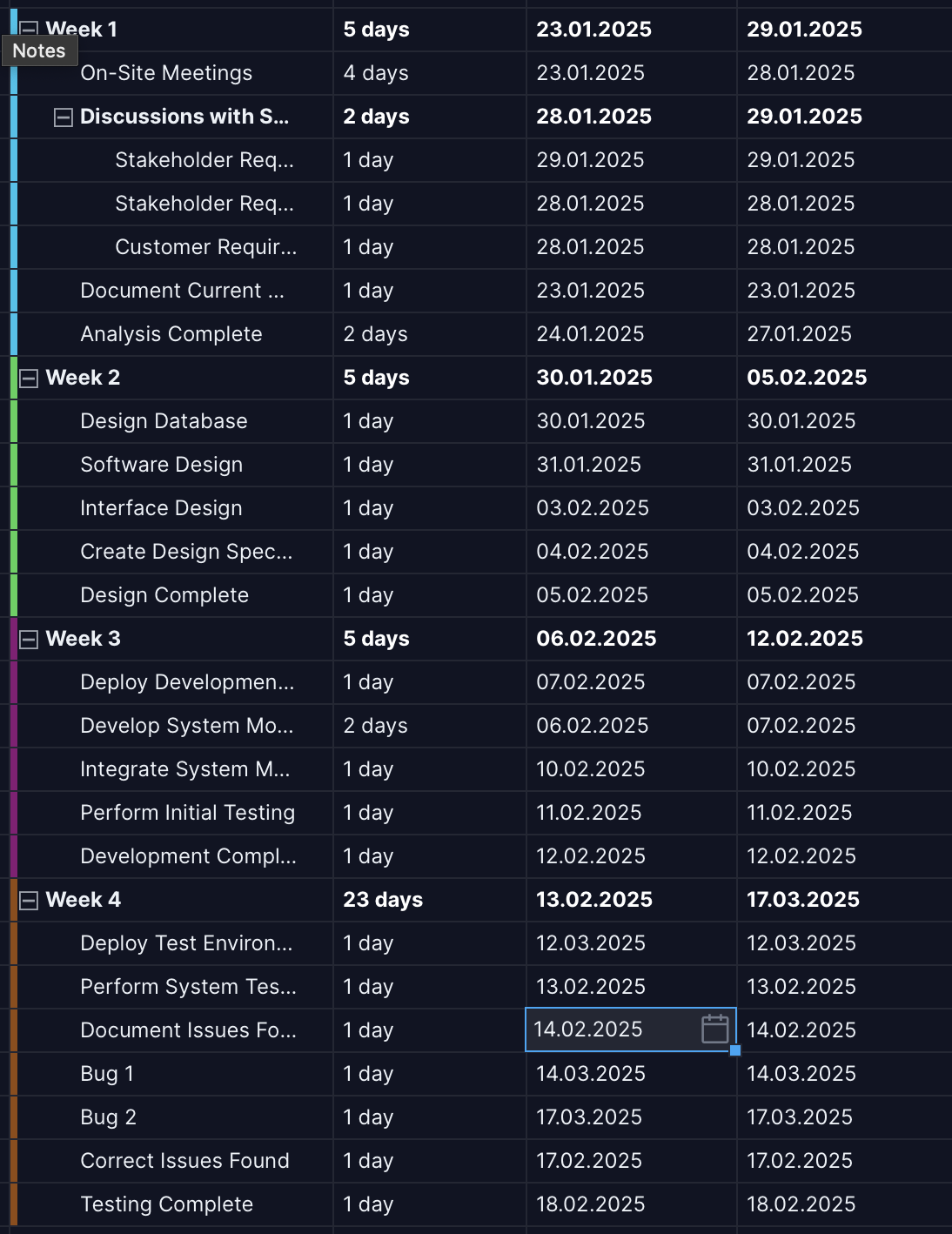
* **Synchronized Calendar:** Ensure real-time updates and synchronization across devices and platforms.
* **Table Templates:** Provide pre-designed templates for common scheduling needs, saving time and enhancing usability.

**Technology Stack**

* **Frontend**:
  + **React.js**: Один из самых популярных фреймворков для создания интерфейсов. Он известен своей эффективностью и совместимостью с бэкенд-системами на GoLang, что делает его идеальным для построения масштабируемых приложений.
  + **Tailwind CSS**: Утилитарный CSS-фреймворк, который помогает быстро создавать адаптивные и красивые интерфейсы с минимальными усилиями. В проекте используется для стилизации компонентов и страниц.
  + **Vite**: Быстрый и современный инструмент для сборки фронтенд-приложений. Он обеспечивает молниеносную сборку и горячую перезагрузку, что ускоряет процесс разработки.
* **Backend**:
  + **GoLang** с **Fiber**: Go — это язык программирования с высокой производительностью, идеально подходящий для обработки большого числа запросов. Fiber — минималистичный и быстрый веб-фреймворк для Go, который улучшает обработку маршрутов и ускоряет разработку API. Это решение идеально подходит для создания высокопроизводительных и масштабируемых приложений, таких как система управления расписанием.
* **Database**:
  + **MongoDB** с **Atlas**: MongoDB — это документно-ориентированная база данных, хорошо подходящая для работы с изменяющимися и сложными данными, такими как расписания. В проекте используется MongoDB Atlas — полностью управляемый облачный сервис, который предоставляет гибкость, масштабируемость, безопасность и автоматические резервные копии.
* **Development Tools**:
  + **Go Modules**: Для управления зависимостями в Go используется Go Modules, что упрощает добавление и обновление библиотек.
  + **React Components**: Компоненты UI для аутентификации, управления пользователями и расписаниями, которые упрощают создание и модульное тестирование интерфейса.
  + **MongoDB Connection**: Коннектор для взаимодействия с MongoDB, обеспечивающий эффективную работу с базой данных через Fiber.

**Development Plan**

* **Platform Development:** Design a user-friendly interface with intuitive navigation and functionality.
* **Database Collections:** Implement collections for:
  + **Users:** Store user data, roles, and preferences.
  + **Tables:** Manage scheduling data.
  + **Teams:** Facilitate collaboration and group management.
* **Integration and Testing:** Ensure seamless synchronization and robust functionality through rigorous testing.

**Schedule  
**  
Week 1 (23.01.2025 – 29.01.2025)   
• On-Site Meetings (4 days) - Assigned to Sayan (Backend - GoLang)   
• Discussions with Stakeholders (3 days)   
• Stakeholder Requirements (1 day) - Assigned to Chingis (Frontend - React.js)   
• Customer Requirements (1 day) - Assigned to Nuraidyn (Database - MongoDB)   
• Document Current Process (4 days) - Assigned to Sayan (Backend - GoLang)   
• Analysis Complete (2 days) - Assigned to Chingis (Frontend - React.js)

Week 2 (30.01.2025 – 05.02.2025)   
• Design Database (4 days) - Assigned to Nuraidyn (Database - MongoDB)   
• Software Design (3 days) - Assigned to Sayan (Backend - GoLang)   
• Interface Design (2 days) - Assigned to Chingis (Frontend - React.js)   
• Create Design Specifications (3 days) - Assigned to Nuraidyn (Database - MongoDB)   
• Design Complete (1 day) - Assigned to Sayan (Backend - GoLang)   
• Deploy Development Environment (1 day) - Assigned to Chingis (Frontend - React.js)

Week 3 (06.02.2025 – 12.02.2025)   
• Develop System Modules (6 days) - Assigned to Nuraidyn (Database - MongoDB)   
• Integrate System Modules (5 days) - Assigned to Sayan (Backend - GoLang)   
• Perform Initial Testing (3 days) - Assigned to Chingis (Frontend - React.js)   
• Development Complete (1 day) - Assigned to Sayan (Backend - GoLang)

Week 4 (13.02.2025 – 17.02.2025)   
• Deploy Test Environment (1 day) - Assigned to Chingis (Frontend - React.js)   
• Perform System Testing (4 days) - Assigned to Nuraidyn (Database - MongoDB)   
• Document Issues Found (3 days) - Assigned to Sayan (Backend - GoLang)   
• Bug Fixing (3 days)   
• Bug 1 (1 day) - Assigned to Chingis (Frontend - React.js)   
• Bug 2 (1 day) - Assigned to Nuraidyn (Database - MongoDB)   
• Correct Issues Found (1 day) - Assigned to Sayan (Backend - GoLang)   
• Testing Complete (1 day) - Assigned to Chingis (Frontend - React.js)

**Conclusion**

The proposed timetable management system bridges the gap between generic scheduling tools and the specific needs of users requiring efficient timetable management. By addressing market gaps and integrating key functionalities, this project offers a valuable solution for individuals and organizations alike.