**Product Requirements Document (PRD)**

**1. Overview**

**Product Name:** Productivity Optimizer Mobile Webapp  
**Purpose:** A personal, mobile-first web application that aggregates data from Todoist and Google Calendar, uses OpenAI to generate insights and productivity plans, sends custom reports via Gmail, and now provides advanced analytics and trend visualizations.  
**Audience:** For personal use only (no user management)  
**Style Inspiration:** A clean, simple interface influenced by the Todoist mobile app, with intuitive bottom navigation and responsive design.

**2. Objectives**

* **Integrate Multiple APIs:** Seamlessly connect with Todoist, Google Calendar, OpenAI, and Gmail.
* **Data Aggregation & Filtering:** Import and parse both completed and active Todoist tasks along with calendar events, with flexible filtering options.
* **Actionable Insights & Planning:** Use OpenAI to generate productivity reports and daily plans based on custom filters.
* **Automated Communication:** Send reports and plans immediately or on schedule via Gmail.
* **Advanced Analytics:** Provide a dedicated “Trends” tab featuring clean, interactive visualizations of both standardized analytics (e.g., weekly completed tasks, daily productivity scores) and custom trends based on user-defined metrics.
* **User Experience:** Offer a mobile-first interface with four main navigation tabs—Plan, Review, Email, and Trends—that provide a cohesive, intuitive user journey.

**3. Scope**

**In Scope**

* **Todoist Integration:**
  + Import tasks (both completed and active) via the Todoist API.
  + Parse essential task data (IDs, content, due dates, labels, status, etc.).
* **Google Calendar Integration:**
  + Import calendar events (upcoming and within custom ranges) via the Google Calendar API.
  + Parse event details such as title, times, and descriptions.
* **OpenAI Integration:**
  + Generate productivity reports (e.g., weekly summaries) and daily productivity plans using custom prompts.
  + Support live chat-style follow-up interactions for deeper insights.
* **Gmail Integration:**
  + Send immediate or scheduled emails containing generated reports and plans using the Gmail API.
  + Manage both one-time and recurring email delivery with customizable frequency settings.
* **User Interface:**
  + Mobile-first responsive design with a clean, minimal look inspired by Todoist.
  + Four navigation tabs: **Plan**, **Review**, **Email**, and **Trends**.
  + Configuration panel for setting date ranges, email frequency, and OpenAI prompt customizations.
  + Save and reuse filter and report templates.
* **Trends Tab (Advanced Analytics):**
  + Display standardized analytics such as tasks completed per week, overdue tasks, and daily productivity scores.
  + Provide interactive, clean visualizations (charts/graphs) for both standard metrics and custom user-defined trends.
  + Allow users to customize the visualization parameters and filter data to uncover unique insights.

**Out of Scope**

* Multi-user authentication or user management (the app is designed for personal use only).
* Integrations beyond Todoist, Google Calendar, OpenAI, and Gmail (unless specified in future enhancements).
* Complex enterprise-level analytics beyond the scope of personal productivity insights.

**4. User Journey & Navigation**

**Overall Navigation**

The application will feature a bottom navigation bar with four tabs:

* **Plan:** Create and generate daily productivity plans using filtered task and event data.
* **Review:** Generate comprehensive productivity reports based on tasks and calendar events.
* **Email:** Manage and send automated or one-time email reports using saved templates.
* **Trends:** View advanced analytics and trend visualizations based on standardized metrics and custom user-defined trends.

**Detailed User Flow**

1. **Landing & Dashboard:**
   * The user lands on a dashboard where the default active tab is shown.
   * Quick access to immediate actions (e.g., “Summarize and Send Email Now”, “Chat Live About Your Tasks”) is provided.
2. **Plan Tab:**
   * **Filtering Interface:** Define custom filters for tasks and calendar events (e.g., select date ranges, task status, labels).
   * **Plan Generation:** Aggregate filtered data and invoke OpenAI to create a daily productivity plan.
   * **Template & Email Options:** Save filter settings as a template and opt to receive the plan via email.
3. **Review Tab:**
   * **Reporting Interface:** Apply filters to tasks (and optionally calendar events) to generate productivity reports.
   * **Report Generation:** Use a custom OpenAI prompt to produce a report summarizing performance over a defined period.
   * **Template & Email Options:** Save report settings and schedule or immediately email the report.
4. **Email Tab:**
   * **Email Management:** Create, edit, and manage email templates.
   * **Scheduling Options:** Set up automated or one-time email deliveries with configurable frequency (Daily, Weekly, Monthly, Custom).
5. **Trends Tab:**
   * **Standardized Analytics:** View pre-built visualizations such as weekly task completion, overdue tasks, and productivity scores.
   * **Custom Trends:** Customize filters and select metrics to generate personalized trend charts.
   * **Interactive Visualizations:** Clean, interactive charts (e.g., bar graphs, line charts, pie charts) that allow zooming, tooltips, and data comparisons.
   * **Export & Share:** Optionally export visualizations or download data for further analysis.
6. **Configuration & Settings:**
   * **Centralized Settings Panel:** Adjust email frequency, default date ranges, OpenAI prompts, and other parameters.
   * **Immediate Actions:** Options for live interaction with OpenAI and immediate report/email sending.

**5. Functional Requirements**

**5.1 Data Import & Parsing**

* **Todoist API:**
  + Import all tasks, including both completed and active tasks.
  + Parse fields: task ID, content, due date, labels, assignee, project info, and completion status.
  + Support both manual and scheduled data refreshes.
* **Google Calendar API:**
  + Import upcoming events and events within user-defined ranges.
  + Parse event details: title, start/end times, location, and description.

**5.2 Data Filtering & Organization**

* **Custom Filters:**
  + Allow definition of custom filters using date pickers, checkboxes, and dropdown menus.
  + Filter by task status, labels, projects, or assignees.
  + Enable saving of filter configurations as templates for reuse in Plan, Review, and Trends tabs.

**5.3 OpenAI Integration**

* **Productivity Report & Daily Plan Generation:**
  + Use filtered task and calendar data as input to custom OpenAI prompts.
  + Generate daily plans and productivity reports.
  + Provide live chat-style follow-up interactions for more detailed insights.

**5.4 Gmail Integration**

* **Email Delivery:**
  + Use the Gmail API to send immediate or scheduled emails with generated content.
  + Support configurable email frequencies (Daily, Weekly, Monthly, Custom).
* **Email Template Management:**
  + Create, save, and edit email templates for reports and plans.

**5.5 Trends & Advanced Analytics**

* **Standard Analytics Visualizations:**
  + Display charts for standardized metrics such as:
    - Weekly task completions.
    - Daily productivity scores.
    - Number and percentage of overdue tasks.
  + Provide visual representations (line charts, bar graphs, pie charts) that are clean and mobile-optimized.
* **Custom Trends:**
  + Allow users to select specific metrics, time frames, and filters to generate custom trend charts.
  + Enable interactive features like tooltip details, zooming, and data export options.
* **Data Integration:**
  + Integrate data from both Todoist and Google Calendar to provide combined trend insights (e.g., correlation between meeting load and task completion rates).

**5.6 User Interface & Experience**

* **Mobile-First Design:**
  + Responsive layout using HTML5, CSS3 (e.g., Bootstrap 5), and JavaScript.
  + Consistent, minimalist design with a clean aesthetic inspired by the Todoist app.
* **Immediate Feedback:**
  + Display loading indicators, success messages, and error notifications for all actions.
* **Navigation:**
  + A fixed bottom navigation bar providing easy access to Plan, Review, Email, and Trends tabs.
* **Visualization Tools:**
  + Use libraries such as Chart.js or D3.js for rendering interactive charts in the Trends tab.

**5.7 Configuration & Settings**

* **Centralized Settings Panel:**
  + Control email frequency, default date ranges, OpenAI prompt customizations, and visualization preferences.
* **Persistence:**
  + Save settings locally or in a lightweight backend (e.g., Google Sheets) to ensure configuration persists across sessions.

**6. Non-Functional Requirements**

**6.1 Performance**

* **Data Fetching & Rendering:**
  + API calls and data parsing should complete within seconds.
  + Visualizations should load quickly and be responsive to user interactions.
* **UI Responsiveness:**
  + Mobile interface must perform smoothly, with fast transitions and minimal latency.

**6.2 Security**

* **API Key & OAuth Management:**
  + Secure handling and storage of API keys and OAuth tokens for all integrated services.
  + Ensure tokens are never exposed to the client side.
* **Data Privacy:**
  + Utilize HTTPS for secure data transmission.
  + Protect any locally stored data via secure storage practices.

**6.3 Scalability & Maintainability**

* **Code Organization:**
  + Use a modular structure that separates API integrations, business logic, and UI components.
* **Ease of Updates:**
  + Allow configuration templates and prompts to be updated without full code redeployment.

**6.4 Cross-Browser and Mobile Compatibility**

* **Responsive Design:**
  + Ensure compatibility with all modern mobile browsers (Chrome, Safari, Firefox) with consistent styling.
* **Progressive Enhancement:**
  + Provide fallback functionality if certain API integrations are temporarily unavailable.

**7. Technical Architecture & Integration Details**

**7.1 Technology Stack**

* **Frontend:**
  + HTML5, CSS3 (Bootstrap 5), JavaScript.
  + Visualization libraries such as Chart.js or D3.js for trends and analytics.
* **Backend / Google Apps Script:**
  + Utilize Google Apps Script for integration with Google APIs (Calendar, Gmail, and Sheets for settings).
  + Serverless functions to handle API calls to Todoist and OpenAI.
* **APIs:**
  + **Todoist API:** RESTful endpoints to fetch and update tasks.
  + **Google Calendar & Gmail API:** OAuth 2.0 authentication and RESTful calls for event handling and email sending.
  + **OpenAI API:** HTTPS calls for generating insights, reports, and plans.

**7.2 Data Flow Diagram**

1. **Data Import:**
   * User initiates data refresh manually or via schedule.
   * Fetch tasks from Todoist and events from Google Calendar.
2. **Filtering & Aggregation:**
   * User applies filters via the Plan, Review, or Trends tabs.
   * Filtered data is cached or temporarily stored for processing.
3. **Processing via OpenAI:**
   * Filtered data is passed to OpenAI with custom prompts for report/plan generation.
   * The generated content is returned and rendered in the appropriate tab.
4. **Email Delivery:**
   * Generated reports or plans are formatted and sent via the Gmail API.
5. **Trends Visualization:**
   * Aggregated data is processed and visualized using interactive charting libraries.
   * Users interact with the charts to view detailed trend insights.

**8. User Interface Design**

**8.1 Overall Look & Feel**

* **Clean, Minimal, and Modern:**
  + Inspired by the Todoist mobile app, with a consistent color scheme and a red accent for primary actions.
* **Responsive & Interactive:**
  + Mobile-first design ensuring smooth interaction, especially in the Trends tab where data visualizations must be clear and engaging.

**8.2 Navigation Layout**

* **Bottom Navigation Bar:**
  + Four tabs: “Plan,” “Review,” “Email,” and “Trends.”
  + Persistent navigation with each tab maintaining its own functional focus.

**8.3 Screen-by-Screen Breakdown**

* **Plan Screen:**
  + Header, filter section, and a “Generate Plan” button.
  + Options to save filter settings as a template and email the plan.
* **Review Screen:**
  + Header with reporting filters.
  + “Generate Report” button and template/email options.
* **Email Screen:**
  + List and management of saved email templates.
  + Scheduling and one-time email options.
* **Trends Screen:**
  + **Standard Visualizations:** Pre-configured charts showing key metrics (e.g., tasks completed per week, overdue tasks).
  + **Custom Visualization Builder:** Interface to select metrics, set time frames, and apply custom filters.
  + **Interactive Elements:** Charts that allow hovering, zooming, and data export.
* **Settings/Configuration Panel:**
  + Options for adjusting email frequency, default date ranges, OpenAI configurations, and visualization preferences.

**9. Testing & Deployment**

**9.1 Testing**

* **Unit Tests:**
  + Test individual modules including API integrations, filter logic, and chart rendering.
* **Integration Tests:**
  + Simulate full workflows from data fetching, filtering, report/plan generation, email sending, and trend visualization.
* **UI/UX Testing:**
  + Validate responsiveness and usability across various mobile devices and browsers.
* **User Acceptance Testing (UAT):**
  + Ensure that all user journeys—Plan, Review, Email, and Trends—function as intended and deliver actionable insights.

**9.2 Deployment**

* **Hosting:**
  + Deploy as a web app via Google Apps Script or a static hosting provider with backend functions.
* **Versioning:**
  + Use a version control system (e.g., Git) to manage codebase iterations.
* **Maintenance:**
  + Monitor API changes and update integration logic; incorporate user feedback to refine visualizations and analytics.