SOFTWARE ANALYSIS DOCUMENT

Project Name: CollabFlow

Category: Real-Time Collaborative Task + DevOps + Product Management Platform

Tech Stack: MERN + Socket.IO

Timeline: 3 Weeks

1. INTRODUCTION

CollabFlow is a **developer-focused productivity, project management, and collaboration platform** modeled after **Monday Dev**.
It supports:

- Software development lifecycle
- Agile / Scrum
- Sprint management
- Bug tracking
- Tasks & dependencies
- Real-time collaboration
- Automation
- Analytics
- Dashboards
- DevOps workflows
- Integrated team collaboration

2. PROJECT OBJECTIVES

1. Build a **Monday Dev–grade system** entirely in pure MERN.
2. Provide real-time updates across all project modules.
3. Offer dev-centric features like sprints, bugs, backlog, releases, etc.
4. Deliver deep analytics and customizable dashboards.
5. Support Agile, Kanban, Scrum, and product roadmap workflows.

3. SCOPE

✔ Sprint Management

✔ Backlog Management

✔ Kanban & Scrum Views

✔ Burndown & Velocity Charts

✔ Cycle-Time Analytics

✔ Bug & Issue Tracking

✔ Product Requirements Board

✔ Gantt Roadmaps

✔ Release Management

✔ Dev Activity Timeline

✔ Cross-Team Collaboration Channels

✔ Automation Workflows (logic-based)

✔ Custom Fields & Custom Views

✔ Reusable Board Templates

✔ Team Workload & Capacity Tracking

✔ Developer Activity Heatmap

✔ Git-like update timeline (without integration)

✔ Advanced dashboards with widgets

All combined with your original features (real-time updates, RBAC, notifications, predictive engine, etc.).

4. DASHBOARD (Monday Dev – Style)

🟦 Main Dashboard Widgets

1. **Sprint Progress Widget**
2. **Burndown Chart**
3. **Velocity Chart**
4. **Open Bugs Counter**
5. **Completed Tasks Counter**
6. **Project Health Indicator**
7. **Team Activity Heatmap**

🟩 Project Timeline (Gantt View)

For roadmap planning & releases.

🟧 Backlog View

Sortable by:

- Priority
- Deadline
- Story points
- Owner
- Status

🟨 Sprint Board (Scrum View)

Columns:

- To Do
- In Progress
- In Review
- Completed

🟥 Bug Tracking Board

Fields:

- Severity (Low/Medium/High/Critical)
- Reproducibility
- Steps to reproduce
- Attach logs/screenshots

🟪 Team Capacity View

Shows:

- Workload distribution
- Overloaded members

- Idle members

5. SYSTEM ARCHITECTURE

Modules:

1. **Agile & Sprint Module**
2. **Backlog Module**
3. **Bug Tracker Module**
4. **Roadmap & Release Module**
5. **Custom Automations Engine**
6. **Custom Boards & Templates Module**

6. FUNCTIONAL REQUIREMENTS

6.1 User Roles (RBAC)

- **Admin**
- **Project Manager**
- **Developer**
- **QA Team Member**
- **Viewer/Stakeholder**

6.2 Core Feature Groups

A. Task & Project Management

- Real-time task updates (Socket.IO)
- Subtasks, attachments, dependencies
- Assignees & watchers
- Mentions and comments

B. Agile Features (Monday Dev Replicated)

1. Sprints

- Create/Edit/Delete sprints
- Assign tasks to sprints
- Velocity tracking
- Burndown chart

2. Backlog

- Fully sortable & filterable
- Drag items into sprints
- Priority indexing

3. Story Points System

- Story points
- Estimate tracking

4. Release Management

- Versions
- Milestones
- Version timelines

5. Dev Timeline

- Git-like task update streams
- Event-based timeline

C. Boards & Views

Available Views:

- Kanban
- Scrum
- Timeline (Gantt)
- Calendar
- Table
- Workload
- Dashboard (Widgets)

Board Features:

- Drag & drop
- Group by status, priority, owner
- Custom fields (text, number, dropdown, date, etc.)
- Board templates
- Save custom views

D. Bug Tracking System (Monday Dev Level)

Fields:

- Title
- Severity
- Status
- Environment
- Reproducible?
- Steps to reproduce
- Attachments

With:

- Real-time bug updates
- Bugs visible in sprint
- Bugs visible in backlog
- Bug metrics dashboard

E. Automation Engine (Logic-Based, No AI)

Users can create rules like:

- "When task status changes to Done → move to Completed group"
- "When due date is tomorrow → notify assignee"
- "When bug severity is Critical → alert entire team"

Pure logic-based, no machine learning needed.

F. Dashboards (Monday Dev Style)

Admins & PMs can add:

- Velocity widget
- Burndown widget
- Open bug counter
- Task progress donut
- Workload distribution
- Heatmap activity chart
- Timeline widget

G. Analytics (Developer Focused)

Includes:

- Cycle Time
- Lead Time
- Average Task Completion Time
- Priority Distribution
- Release Forecasting
- Sprint Efficiency Score

H. Notifications

- Email
- In-app
- Real-time popups
- Automation-triggered notifications

- Prioritized notifications (Critical > Medium > Low)

7. NON-FUNCTIONAL REQUIREMENTS

Performance

- Handle 10,000+ tasks per sprint
- Real-time updates < 150ms
- Up to 10,000 users online with Socket.IO rooms

Scalability

- Micro-modular MERN architecture
- Refined indexing for backlog, sprints, and bug DB collections

8. DATABASE DESIGN

New Collections

Sprints

```
{
  _id,
  projectId,
  name,
  startDate,
  endDate,
  status,
  velocity,
  createdAt
}
```

Backlog

```
{
 _id,
 projectId,
 items: [taskIds],
 createdAt
}
```

Bug Reports

```
{
 _id,
 taskId,
 severity,
 reproducible,
 steps,
 attachments: [],
 status,
 createdBy,
 createdAt
}
```

Releases

```
{
 _id,
 projectId,
 version,
 description,
 releaseDate,
 tasks: [taskIds]
}
```

Custom Board Templates

```
{
 _id,
 createdBy,
 name,
 fields: [...],
 layoutConfig: {...}
}
```

9. DELIVERABLES

You will now deliver a system equivalent to **Monday Dev**, including:

- Sprint management
- Backlog
- Issues/Bugs board
- Gantt timelines
- Dev analytics
- Dashboard widgets
- Automations
- Custom workflows
- Templates
- Real-time engine