Agile Team Capacity Tracking with GitHub Integration

Group Members:

- **Muzammil Ahmad**
- **Abdullah Amjad**
- **Abdul Moiz**
- **Abdul Wasay**
- **Daniyal Gul**

Submitted To: Dr. Nabeel Awan

Submission Date: 27th April 2025

Agile Team Capacity Tracking with GitHub Integration

1. Project Overview

The Agile Team Capacity Tracking System with GitHub Integration is designed to help Agile teams effectively monitor, analyze, and manage their workload based on real-time GitHub activity.
It enables project managers and developers to track team capacity, sprint progress, pull requests, commits, and issue activity, ensuring better planning and workload distribution.

---

2. Key Features

- **Real-Time GitHub Integration**
  Automatically fetches and updates data on pull requests, commits, and issues.
- **Dashboard Visualization**
  A clean, interactive dashboard to visualize team activity and sprint progress.
- **Team Capacity Monitoring**
  Helps identify bottlenecks by tracking open issues, pending reviews, and developer contributions.
- **API Backend**
  A simple Java backend providing RESTful APIs for frontend consumption.
- **Automation Support**
  Scheduled fetching and updating of GitHub data for continuous monitoring.

---

3. System Architecture

- **Frontend:** Html, CSS Dashboard
- **Backend:** Java Spring Boot APIs
- **GitHub Integration:** GitHub REST API

- **Database (Optional):** Local in-memory or cloud database for persistence
- **Deployment:** Local development with Docker (optional for production)

---

4. Components

4.1 Frontend (HTML, CSS)

- Displays:
    - Sprint status (open pull requests, open issues)
    - Commits per developer
    - Issue assignments
- Components:
    - Pull Requests Summary
    - Commits Summary
    - Issues Summary
    - Sprint Capacity Overview

4.2 Backend (Java Spring Boot)

- REST APIs:
    - /api/pullrequests → Fetch list of active pull requests
    - /api/commits → Fetch recent commits by team members
    - /api/issues → Fetch open issues
- GitHub API Client:
    - Authenticates with GitHub using a personal access token
    - Handles rate limits and retries

4.3 GitHub Integration

- Authentication: Personal Access Token (PAT)
- Data fetched:
    - Repository Pull Requests

    - Repository Commits
    - Repository Issues
- API Rate Handling: Monitored to avoid interruptions

---

5. Technologies Used

Layer	Technology
Frontend	Html, CSS
Backend	Java Spring Boot
API Client	GitHub REST API v3
Deployment	Docker (optional)

7. Future Improvements

- Authentication and role-based access for multiple teams.
- Sprint and workload prediction using historical GitHub data (AI/ML).
- Slack / MS Teams notification integration.
- GitHub Actions-based automation for updates.

---

8. Conclusion

This project provides a simple yet effective way to monitor Agile team capacity by leveraging GitHub activity.
It aims to enhance team transparency, improve sprint planning, and proactively detect capacity issues.

The modular architecture allows future enhancements like multi-repo tracking, prediction analytics, and integration with project management tools (e.g., Jira, Trello).