SPRINT 1 RETROSPECTIVE MEETING MINUTES

DATE:	START TIME:	END TIME:	MEETING PLATFORM:
November 02, 2024	10:30 PM	11:15 PM	Google Meet

ATTENDEES: Sadia(SH), Sumaiya(UJ), Jannnati Tajrimin(TM), Trisha Sarkar(TS), Rubayed All Islam(RAI)

MEETING CHAIR AND NOTE_TAKER: Akila Nipo (AN)

SPRINT 1 EVALUATION IN RETROSPECT

- **What went well?**
- **What went wrong?**
- **How can we improve?**
- **What have we learnt?**

WHAT WENT WELL		WHAT WENT WRONG		HOW TO IMPROVE
Upload Files: **1.2Successfully stored Information.** **2.Successfully created tables in database for users and others (etc, syllabus).** **3.Grant the valid xml and csv format files**: Csv for user files and xml for syllabus info.		Problems faced during unit testing. Some test cases are failed.		Will recheck and correct the issue for unit testing. Will generate separate test cases for unit testing .
Assign Course Teacher: Assigned Course Teacher by clicking on update button to each of the course in an Exam Year. Also assigned course teacher by uploading a CSV file.		Problem faced while updating Exam Committee Member to each of the exam year.		May be mapping problem. Further improvement may solve the problem.

View Class Routine: View Class Routine on clicking the search button went smoothly.		Problem fetched for fetching Class Routine data from Database.		Further recheck and Improvements may solve the problem.

WHAT WENT WELL	WHAT WENT WRONG	HOW TO IMPROVE
Generate Routine: 1. **Dynamic Slot Creation with Lunch Breaks:** 2. **Database-Driven Scheduling Parameters:** Routine algorithm accurately retrieved teacher availability, assigned courses, preferred days/times, room categories, and room requirements from the database for generating routine. 3. **Lunch Break Conflict Avoidance:** Slots were allocated to avoid overlaps during lunch hours, ensuring a smooth schedule without interruptions. 4. **Course-Specific Slot Distribution (Lab/Theory):** Slots were reserved based on course type, with lab sessions receiving double slots as needed 5. **Room Specific Course Distributions** **6. Course Distributions according to counts per week**	1. **Routine Start Delayed by One Slot:** Scheduling started one slot later than the user-defined time. 2. **Underutilization of Slots and Rooms:** To avoid conflicts, some slots and rooms remained unused. 3. **Failed JSON Parsing for Complex Test Cases:** Test cases with arrays of objects in JSON weren't read correctly. 4. **Further improvement of test cases**	1. **Start Routine Exactly at User-Specified Time:** Adjust the logic to ensure scheduling begins precisely at the start time given by the user. 2. **Improve Slot and Room Utilization:** Refine the allocation approach to make better use of available slots and rooms while avoiding conflicts. 3. **Enhance JSON Parsing for Complex Structures:** Strengthen JSON parsing to handle arrays of objects, ensuring all test cases load correctly. 4. **Implement Real-Time Conflict Checks:** Add validation to detect and resolve conflicts during scheduling

View Personalized Dashboard: Made Dashboard for all the users of the system (Teacher, Student, Staff, SuperUser)		Problem faced while updating teacher's Schedule class and Preferred slot onto the navbar		Further recheck and Improvements have solved the problem.
View Class Representative Information: View CR's details such as Name, contact Info. (email, phone no.) clicking on the CR's Dashboard.		Problem for fetching or Updating Class Representative data from Database.		Further recheck and Improvements have solved the problem

What have we learnt?

In Sprint 1, we gained valuable insights and practical skills that enhanced our project management and collaboration. Here are the key takeaways:

1. **Effective Use of Project Management Tools**
   - We utilized **Trello** for task organization, allowing us to visualize progress and manage workloads effectively.
   - **Discord** served as a platform for real-time communication, facilitating discussions and quick decision-making among team members.
   - **GitHub** was essential for version control, enabling collaborative coding efforts and maintaining a history of changes.
   - We employed **Jira** to document and track issues, which helped us prioritize tasks and manage workflow efficiently.
2. **Time Management**
   - The implementation of **Toggl** allowed us to track time spent on various activities, providing insights into our productivity and helping us identify areas for improvement.
3. **Documentation**
   - We adopted **JSDoc** for code documentation, improving the clarity and accessibility of our codebase, which is crucial for both current team members and future contributors.

4. **Unit Testing**
    - The use of **Mocha** for unit testing was beneficial in ensuring code quality, as it enabled us to identify and rectify issues early in the development process.
5. **Meeting Management**
    - Regular meetings were conducted to ensure alignment on project goals and to address challenges collaboratively. This practice fostered better communication within the team.
6. **Team Coordination**
    - Overall, we improved our coordination as a team, clearly defining roles and responsibilities, which contributed to enhanced efficiency and productivity.