Department of Computer Science Engineering

SRM IST, Kattankulathur – 603 203

18CSC206J – SOFTWARE ENGINEERING AND PROJECT MANAGEMENT

Experiment No	05
Title of Experiment	Prepare Work breakdown structure, Timeline chart, Risk identification table
Name of the Candidate	Sai Rohit P
Team Members	Sai Rohit (RA2111003010806) Pavan Sagar (RA2111003010809)
Date of Experiment	

Mark Split Up			
S.No	**Description**	**Maximum Mark**	**Mark Obtained**
1	Exercise	5	
2	Viva	5	
Total		**10**	

Staff Signature with date

Aim:

To Prepare Work breakdown structure, Timeline chart and Risk identification table

Team Members:

S. No.	Register Number	Name	Role
1	RA2111003010806	Sai Rohit	Rep
2	RA2111003010809	Pavan Sagar	Member

Project Title: TimetableSOS

Work Breakdown Structure (WBS) Chart:

1. Project Management
   1.1. Define project scope and objectives
   1.2. Develop project plan and schedule
   1.3. Identify project risks and mitigation strategies
   1.4. Allocate project resources and budget
   1.5. Establish project governance and communication plan
   1.6. Monitor project progress and adjust plan as needed
   1.7. Close out project and conduct project review
2. Requirements Gathering
   2.1. Conduct user research to understand target audience
   2.2. Identify user needs and requirements for the app
   2.3. Determine app features and functionalities based on requirements
   2.4. Create user stories and use cases for the app
   2.5. Finalize requirements documentation and gain stakeholder approval

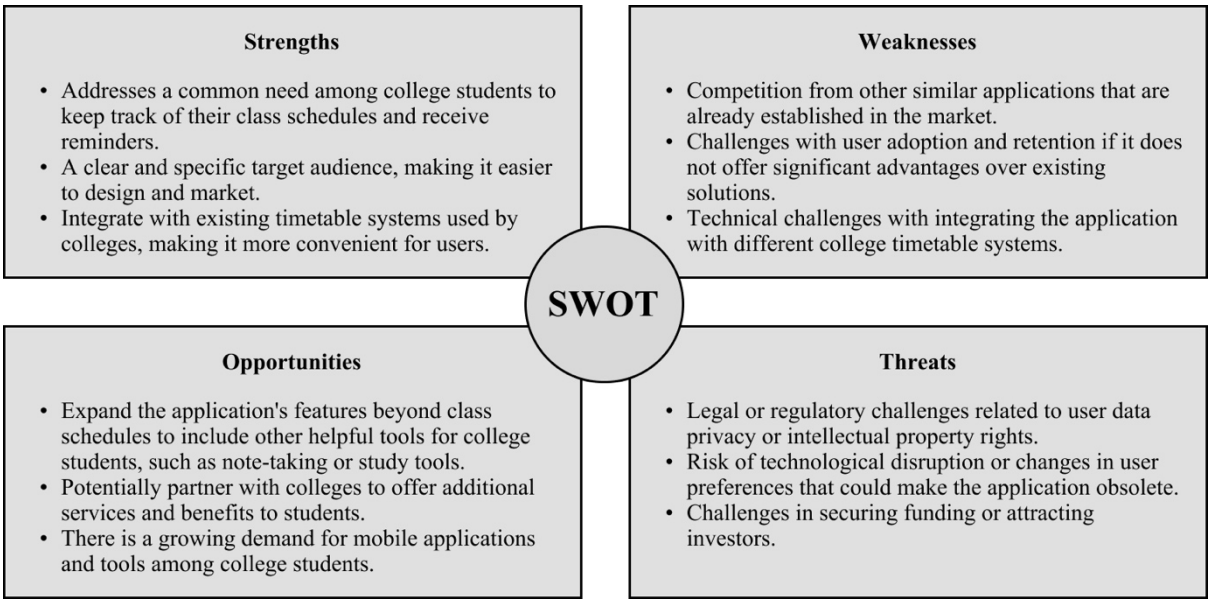
3. Design
    3.1. Develop wireframes and user interface design for the app
    3.2. Design database schema for storing user timetables and notifications
    3.3. Define system architecture and integration points
    3.4. Finalize design documentation and gain stakeholder approval
4. Development
    4.1. Set up development environment and toolchain
    4.2. Develop front-end functionality for the app
    4.3. Develop back-end functionality for the app, including notification logic
    4.4. Integrate front-end and back-end functionality
    4.5. Conduct unit testing of code modules 4.6 Conduct integration testing of the app
    4.6. Fix bugs and issues as identified during testing
5. Quality Assurance
    5.1. Create test cases for functional testing
    5.2. Conduct functional testing of the app, including user flows and notifications
    5.3. Conduct regression testing of the app after bug fixes
    5.4. Conduct user acceptance testing to ensure the app meets stakeholder expectations
    5.5. Conduct performance testing to ensure the app can handle high usage
    5.6. Fix bugs and issues as identified during testing
6. Deployment and Launch
    6.1. Prepare production environment for the app
    6.2. Deploy the app to production servers and cloud services
    6.3. Conduct final testing of the app in production
    6.4. Train users on how to use the app effectively
    6.5. Launch the app to the public and promote its availability
7. Maintenance and Support
    7.1. Provide ongoing technical support to users of the app
    7.2. Conduct periodic evaluations of the app's performance and usage
    7.3. Conduct software updates and maintenance to address issues and add new features
    7.4. Monitor user feedback and address any critical issues promptly
    7.5. Plan for future enhancements and updates to the app based on user feedback and changing requirements

Gantt Chart:

Task	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8
Project Management								
Define project scope and objectives								
Develop project plan and schedule								
Identify project risks and mitigation strategies								
Allocate project resources and budget								
Establish project governance and communication								
Monitor project progress and adjust plan								
Close out project and conduct project review								
Requirements Gathering								
Conduct user research to understand target audience								
Identify user needs and requirements for the ap								
Determine app features and functionalities								
Create user stories and use cases for the app								
Finalize requirements documentation								
Gain stakeholder approval								
Design								
Develop wireframes and user interface design								
Design database schema for storing user timetables								
Define system architecture and integration points								
Finalize design documentation								
Gain stakeholder approval								

Risk Analysis:

SWOT:



RMMM:

Risk	Mitigation	Monitoring	Management
Technical issues with integrating with college timetable systems	Conduct thorough research to ensure compatibility with a wide range of systems	Regularly test the application with different timetable systems	Assign a dedicated technical team to oversee integration and troubleshooting efforts
Insufficient user adoption	Conduct extensive user research to ensure the application meets the needs of the target audience	Gather user feedback regularly and incorporate it into development	Implement a targeted marketing and user engagement strategy to increase adoption.
Data privacy concerns	Implement strong security measures and encryption protocols to protect user data	Conduct regular security audits and vulnerability assessments	Comply with relevant data privacy laws and regulations
Changes in user preferences or technological disruption	Keep up-to-date with industry trends and anticipate changes in user behavior	Conduct regular user surveys and usability tests to ensure the application remains relevant and user-friendly	Develop a plan to adapt to changes or disruptions.
Insufficient funding or investment	Develop a detailed financial plan and budget	Identify potential investors and present a compelling business case	mplement cost-saving measures and explore alternative funding sources.

Result: Thus, the work breakdown structure with timeline chart and risk table were formulated successfully.