LAB REPORT

Submitted by

PRANEETH NARRA [RA2011003010012]

Under the Guidance of

Dr. L. Jayakumar

Associate Professor, Department of Computing Technologies

In partial satisfaction of the requirements for the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE ENGINEERING

with specialization in CSE CORE

SCHOOL OF COMPUTING

COLLEGE OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR - 603203

JUNE 2022

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be **University** *u/s 3 of UGC Act, 1956*

SRM INSTITUTION OF SCIENCE AND TECHNOLOGY
KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this lab report titled "18CSC206J- Software Engineering and Project Management Laboratory" is the bonafide work done by PRANEETH NARRA (RA2011003010012) who carried out the lab exercises under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other work.

SIGNATURE
Dr. L.Jayakumar
SEPM – Course Faculty
Associate Professor
Department of Computing Technologies

Table of contents

ABSTRACT

Now, this project schedules your links according to your meeting.You don’t have to copy any link or use any other app for that as this project will schedule it for you this way you will save your time as well.Storage consumption will be less as you no longer need multiple apps.

This project aims to make your life a bit easier. In these times, when everything is being digital from studying from home to working from home. For that you have separate links for every meeting and managing those links is a bit difficult. Instead, this project will do that for you, all your meetings with specific links will be managed by this project for you and you will no longer need to use multiple apps for that.

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	1
Title of Experiment	To identify the Software Project, Create Business Case, Arrive at a Problem Statement
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA, BHUVANESH E S, PRANEETH NARRA, K CHANDRA KIRAN
Register Number	RA2011003010015, RA2011003010021, RA2011003010012, RA2011003010020
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 Frame a project team, analyze and identify a Software project. To create a business case and Arrive at a Problem Statement for the Link Scheduler

Team Members:

S. No	Register No	Name	Role
1	**RA2011003010015**	**AKKASH ANUMALA**	**Lead/Rep**
2	**RA2011003010021**	**BHUVANESH E S**	**Member**
3	**RA2011003010012**	**PRANEETH NARRA**	**Member**
4	**RA2011003010020**	**K CHANDRA KIRAN**	**Member**

Project Title: LINK SCHEDULER

Project Description

This project aims to make your life a bit easier. In these times, when everything is being digital from studying from home to working from home. For that you have separate links for every meeting and managing those links is a bit difficult, as u need to refer different apps every time for each meeting. Instead, this project will do that for you, all your meetings which had been scheduled with specific links will be managed by this project for you and you will no longer need to refer multiple apps.

Business case

Attached

Result

Thus, the project team formed, the project is described, the business case was prepared and the problem statement was arrived.

ONE PAGE BUSINESS CASE TEMPLATE

DATE	
SUBMITTED BY	AKKASH ANUMALA, BHUVANESH E S, PRANEETH NARRA, K CHANDRA KIRAN
TITLE / ROLE	Link Scheduler

LOGO

THE PROJECT

In bullet points, describe the problem this project aims to solve or the opportunity it aims to develop.

- In these times, when everything is being digital from studying from home to working from home.
- For that you have separate links for every meeting and managing those links is a bit difficult, as u need to refer different apps every time for each meeting.
- This app will help us manage our links

THE HISTORY

In bullet points, describe the current situation.

- There is a app called events in the android in which can schedule the programs and events.
- But you can't share the schedule to all the members who are participating in the program event.

LIMITATIONS

List what could prevent the success of the project, such as the need for expensive equipment, bad weather, lack of special training, etc.

In the upcoming days , some Mnc's may upgrade their application and come with the same idea or a better version .

APPROACH

List what is needed to complete the project.

We need a good backend support to manually update the context and scheduling the meeting

BENEFITS

In bullet points, list the benefits that this project will bring to the organization.

- Now, this project schedules your links according to your meeting
- You don't have to copy any link or use any other app for that as this project will schedule it for you this way you will save your time as well.
- Storage consumption will be less as you no longer need multiple apps.

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	2
Title of Experiment	Identification of Process Methodology and Stakeholder Description
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA , BHUVANESH E S, PRANEETH NARRA, K CHANDRA KIRAN
Register Number	RA2011003010015, RA2011003010021, RA2011003010012, RA2011003010020
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 identify the appropriate Process Model for the project and prepare Stakeholder and User Description.

Team Members:

Sl No	**Register No**	**Name**	**Role**
1	**RA2011003010015**	**AKKASH ANUMALA**	**Rep/Member**
2	**RA2011003010021**	**BHUVANESH E S**	**Member**
3	**RA2011003010012**	**PRANEETH NARRA**	**Member**
4	**RA2011003010020**	**K CHANDRA KIRAN**	**Member**

Project Title: Link scheduler

Selection of Methodology :

Incorporate ***Identification of Project Methodology and Stakeholder Description template***

WATERFALL MODEL:

- The Waterfall Model was the first Process Model to be introduced.
- It is also referred to as a linear sequential life cycle model.
- It is very simple to understand and use.
- In a waterfall model, each Phase must be completed before the next phase can begin and there is no overlapping in the phases.
- Waterfall model is the earliest SDLC approach that was used for software development.

To develop the Link Scheduler ,There we decided five different phases

1.Planning
2.Modeling
3.Construction
4.Testing
5.Deployment

1.Planning:

To plan how to develop the application, the features for the application, how to approach the problem statement.

2.Modeling:

Making the pseudocode, design the blueprint or the model of the application.

3.construction:

Coding the problem statement, making it into an application With all the predefined features, simple implementation.

4.Testing:

Test the code in all parameters, debugging the code also testing all the features of application.

5.Deployment:

Launching the application and getting feedback.

Stakeholder description:

Primary Stakeholders :

The primary stakeholders include the following individuals and groups:

- Students, teachers of all classes
- All organizations
- In software field
- In management field
- All the fields where there is will be regular meetings/scheduled meetings

Secondary stakeholders :

It can be used by every individual who will attempt meetings moreover who wills to open a website everyday particularly at a time.

Interest and Influence Matrix :

Interest	Influence
High	High
Low	Low
Low	High
High	Low

Low interest , High influence Students , Teachers	High interest , High Influence Managing People
Low interest , low influence Common people	High interest , Low influence Common people

Stakeholder Name	Activity/ Area /Phase	Interest	Influence	Priority (High/ Medium/ Low)
Managers , leaders , software etc …	Having regular meetings	High	High	High
Common people	Having meetings	Low	High	Medium

Result

Thus the Project Methodology was identified and the stakeholders were described.

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

Department Of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	3
Title of Experiment	System, Functional and Non-Functional Requirements of the Project
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA , BHUVANESH E S , PRANEETH NARRA , K CHANDRA KIRAN
Register Number	RA2011003010015 , RA2011003010021 , RA2011003010012 , RA2011003010020
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 identify the system, functional and non-functional requirements for the project.

Team Members:

S No	Register No	Name	Role
1	RA2011003010015	AKKASH ANUMALA	Rep/Member
2	RA2011003010021	BHUVANESH E S	Member
3	RA2011003010012	PRANEETH NARRA	Member
4	RA2011003010020	K CHANDRA KIRAN	Member

Project Title: Link scheduler

System Requirements :

Requirement	Requirement Specification	Department	Name of Business User / Project Team Member
IR1	Development Machine with 6 GB Ram and 4 Cores	hardware	Developer & Tester
IR2	Code Repository	software	Developer & Tester
IR3	Python shell	software	Developer & Tester

Functional Requirements :

Requirement	**Requirement Specification**	**Department**	**Name of Business User**
E1FR1	Information of the user	Authentication	User
E1FR2	ask for user’s schedule and timings	Registration	User
E1FR3	Scheduling the links according to schedule	Scheduling	User
E1FR4	Opening of links according to timings	Execution	User
E1FR5	Feedback and support	Feedback	User

Non-Functional Requirements :

Requirement	**Category of NFR**	**Requirement Specification**	**Department**	**Name Of Business User**
NFR1	Performance	All pages should load within 3 seconds	Response time & compliance	Users(Buyers & sellers)
NFR2	Performance	Search should bring the results less than 7 seconds	Response time & compliance	Users(Buyers & sellers)
	Availability	Application should be available for 24x7	Availability	Users(Buyers & sellers)
	Scalability	Registration Service should scale to serve 1000 request per second over 5 minutes timespan	Performance & Response time	Users(Buyers & sellers)

	Confidentiality	Each user should have a unique login credential that will take him to his own account	Authentication & Security	Users(Buyers & sellers)
	Compliance	user’s information will be secured and will be confidential.	Authentication & Security	Users(Buyers & sellers)
E1NFR2	Usability	The website should be easy to use and understand	Accessibility	Users(Buyers & sellers)
	Security	user’s identification will be required to enter into any meeting	Authentication & Security	Buyers
	Flexibility	once entered login credentials user can save login credentials.	Flexibility & Support	Buyers
E1NFR1	Reliability	meetings are scheduled without any error	support & feedback	Users(Buyers & sellers)
	Rapidity	meeting is scheduled on time without any delay	support & feedback	Users(Buyers & sellers)

Result

Thus the requirements were identified and accordingly described.

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	4
Title of Experiment	Prepare Project Plan based on scope, Calculate Project effort based on resources and Job roles and responsibilities
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA , BHUVANESH E S , PRANEETH NARRA , K CHANDRA KIRAN
Register Number	RA2011003010015 , RA2011003010021 , RA2011003010012 , RA2011003010020
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 Project Plan based on scope, Calculate Project effort based on resources, Find Job roles and responsibilities

Team Members:

Sl No	Register No	Name	Role
1	RA2011003010015	AKKASH ANUMALA	Lead
2	RA2011003010021	BHUVANESH E S	Member
3	RA2011003010012	PRANEETH NARRA	Member
4	RA2011003010020	K CHANDRA KIRAN	Member

1. Executive Summary

Link Scheduler application mainly aims to make online meetings a bit easier and less time-consuming for the users with an online application that schedules the meeting links according to the user's schedule and makes it organized and hassle-free.

2. Project Management Plan

Describe the key issues driving the project. Summarize the results of the project identification stage (e.g. feasibility assessment and business case). Summarize the solution selected from the Business Case. Define the objectives of the project and the intended business results. Define quantitative and measurable objectives that can be used as criteria by which key stakeholders will judge the success of the project. Some of this information can be extracted from the project charter.

Focus Area	Details
Integration Management	Governance Framework Project Team Structure Roles & Responsibilities of Team Change Management (Change Control, Issue Management) Project Closure
Scope Management	Scope Statement Requirement Management (Gathering, Control, Assumption, Constraint Stakeholder) Define Deliverable Requirement Change Control Activities and Sub-Tasks
Schedule Management	Define Milestones Schedule Control
Cost Management	Estimate Effort Assign Team Budget Control
Quality Management	Quality Assurance: Quality assurance will be managed including governance, roles and responsibilities, tools and techniques and reporting Quality Control: Specify the mechanisms to be used to measure and control the quality of the work products
Resource Management	Estimate and Manage the need People: People & Skills Required Finance: Budget Required Physical: Facilities, IT Infrastructure
Stakeholder	Identifying, Analyzing, Engaging Stakeholders
Communication Management	Determine communication requirements, roles and responsibilities, tools and techniques. [Type of Communication, Schedule, Mechanism Recipient]
Risk Management	Identifying, analysing, and prioritizing project risks
Procurement Management	Adhering to organization procurement process

1. Estimation

1.1. Effort and Cost Estimation

Activity Description	Sub-Task	Sub-Task Description	Effort (in hours)	Cost in INR
Design the user screen	E1R1A1T1 (Effort-Requirement-Activity-Task)	To create a desktop-level website using PYTHON	3	Full Stack-85K Frontend-50K Backend-45K
Research and Development	E1R1A1T2	Evaluation of software technology trends and incorporating them with updates and patches	4	For Research - 80K For Rolling out patches/updates – 50K
Data analytics	E1R1A1T3	Administrator of Data, Database Management, Evaluating trends in data	10 with shifts	Data Analyst- 100K Data Scientist- 90K Database Manager-80K
Customer care	E1R1A1T4	Customer care	24 with shifts	Customer Care-12K to 20K
Application maintenance	E1R1A1T5	Advertisements, Public Relations	6 hours a week	Application developer-10K
Marketing	E1R1A1T6	Advertisements, Public Relations	8-10 On requirement	80k

Effort (hr)	Cost (INR)
1	500

1.2. Infrastructure/Resource Cost [CapEx]

Infrastructure Requirement	Qty	Cost per qty	Cost per item
IR1 : Domain rights	1	2 – 5k	2 – 5k
IR2 : Server firm and other hardware	1	150k	150k
IR3 : Legal and other government documents	1	200k	200k

2.3 Maintenance and Support Cost [OpEx]

Category	Details	Qty	Cost per qty per annum	Cost per item
People	Network, System, Middleware and DB admin Developer , Support Consultant	3	2,000,000	6,000,000
License	Operating System Database Middleware IDE	10	10000	100,000
Infrastructures	Server, Storage and Network	20	20000	400,000

2. Project Team Formation

2.1. Identification Team members

Name	Role	Responsibilities
Akkash	Key Business User (Product Owner)	Provide clear business and user requirements
Praneeth	Project Manager	Manage the project
Akkash	Business Analyst	Discuss and Document Requirements
Bhuvanesh	Technical Lead	Design the end-to-end architecture
Chandra Kiran	UX Designer	Design the user experience
Bhuvanesh	Frontend Developer	Develop user interface
Praneeth	Backend Developer	Design, Develop and Unit Test Services/API/DB
Akkash	Cloud Architect	Design the cost effective, highly available and scalable architecture
Chandra Kiran	Cloud Operations	Provision required Services
Bhuvanesh	Tester	Define Test Cases and Perform Testing

2.2. Responsibility Assignment Matrix

RACI Matrix	Team Members			
Activity	**Name (BA)**	**Name (Developer)**	**Name (Project Manager)**	**Key Business User**
User Requirement Documentation	A	C/I	I	R

A	Accountable
R	Responsible
C	Consult
I	Inform

Reference

1. https://www.pmi.org/
2. https://www.projectmanagement.com/
3. https://www.tpsgc-pwgsc.gc.ca/biens-property/sngp-npms/ti-it/ervcpgpm-dsfvpmpt-eng.html

Result:

Thus, the Project Plan was documented successfully.

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	5
Title of Experiment	Prepare Work breakdown structure, Timeline chart, Risk identification table
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA, BHUVANESH E S, PRANEETH NARRA, K CHANDRA KIRAN
Register Number	RA2011003010015, RA2011003010021, RA2011003010012, RA2011003010020
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:

Sl No	Register No	Name	Role
1	RA2011003010015	AKKASH ANUMALA	Rep
2	RA2011003010021	BHUVANESH E S	Member
3	RA2011003010012	PRANEETH NARRA	Member
4	RA2011003010020	K CHANDRA KIRAN	Member

WBS With Project Schedule

Module(#)	Activity(#)	Assignee	Planned Start Date	Planned End Date	Actual Start Date	Actual End Date	Status
M1	User information	Kiran	27-03-2022	6-04-2022	28-03-2022	6-04-2022	completed
M2	Schedule and timings	Praneeth	29-03-2022	10-04-2022	2-04-2022	29-04-2022	completed
M3	Scheduling links (acc to timings)	Akkash	07-04-2022	18-04-2022	09-04-2022	20-04-2022	completed
M4	Opening Links	Bhuvanesh	20-04-2022	1-05-2022	22-04-2022	29-04-2022	completed
M5	User feedback	Bhuvanesh	27-04-2022	19-05-2022	2-05-2022	27-05-2022	completed

Risk Identification :

- Structured Brainstorming with team and stakeholders
- Checklist is a list of actions/points to be considered [Information can be used from the similar previous projects]
- Risk can be identified from

- Assumption-Constraint analysis
- SWOT Analysis [Strength/Weakness/Opportunity/Threat]

List (Describe) Register

Risk ID (#)	Risk Description	Impact Description
RO1	Software error	Can,t open the links
RO2	Personal Error	Mistakes done by individuals
RO3	Usage may be decreased	Loss

Managing Risk

Risk ID (#)	Status [open/closed]	Risk Appetite[accept/mitigate/ Transfer/Avoid	Action	Action Owner	Target Date	Remarks
RO1	open	Mitigate	upgrading	-	-	-
RO2	open	Avoid	Messaging error	-	-	-
RO3	open	Avoid	-	-	-	-

Result:

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

Department of Networking and Communications

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	6
Title of Experiment	Design a System Architecture, Use Case and Class Diagram
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA, BHUVANESH E S, PRANEETH NARRA, K CHANDRA KIRAN
Register Number	RA2011003010015, RA2011003010021, RA2011003010012, RA2011003010020
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

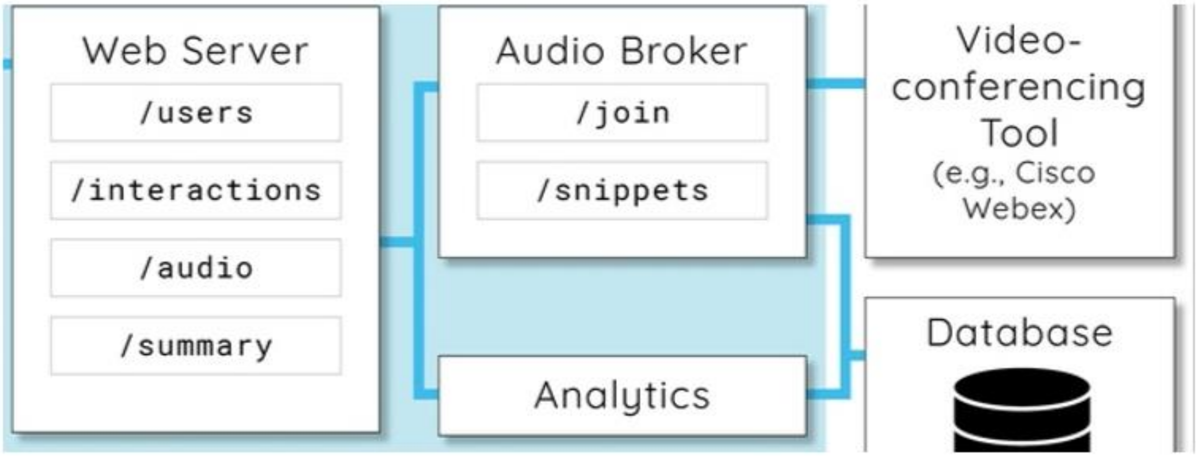
To Design a System Architecture, Use case and Class Diagram

Team Members:

Sl No	Register No	Name	Role
1	RA2011003010015	AKKASH ANUMALA	Rep
2	RA2011003010021	BHUVANESH E S	Member
3	RA2011003010012	PRANEETH NARRA	Member
4	RA2011003010020	K CHANDRA KIRAN	Member

Architecture Diagram with description

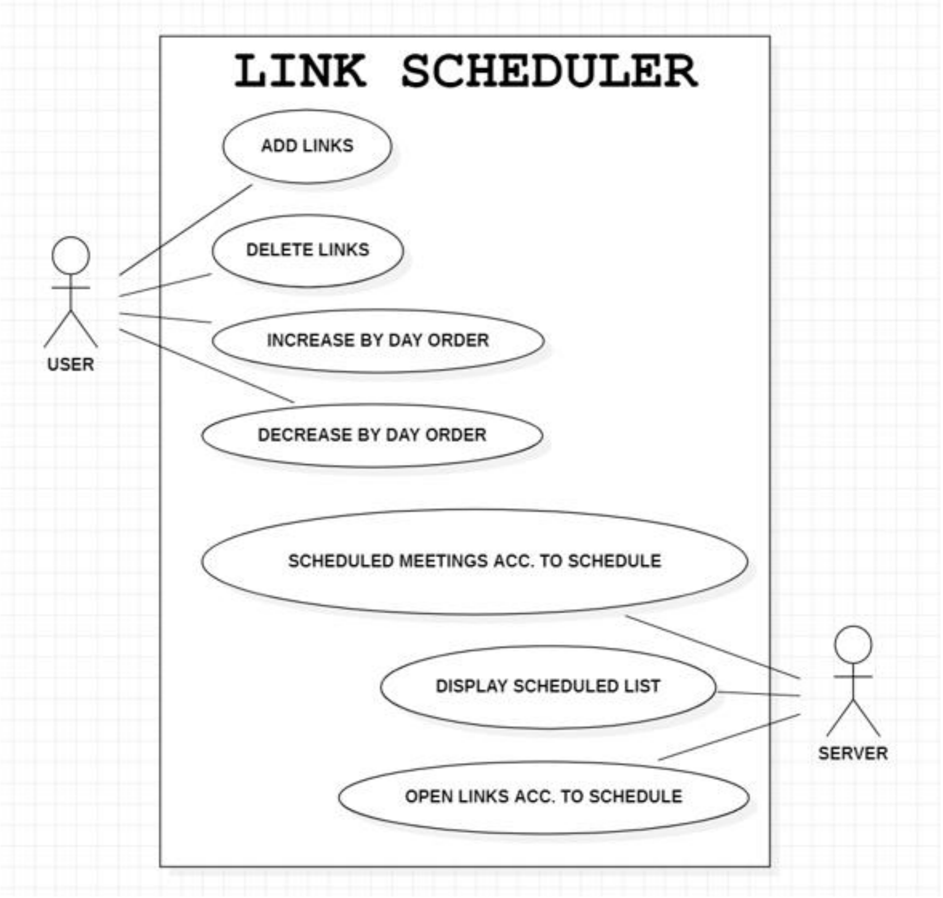
An architectural diagram is a diagram of a system that is used to abstract the overall outline of the software system and the relationships, constraints, and boundaries between components. It is an important tool as it provides an overall view of the physical deployment of the software system and its evolution roadmap.



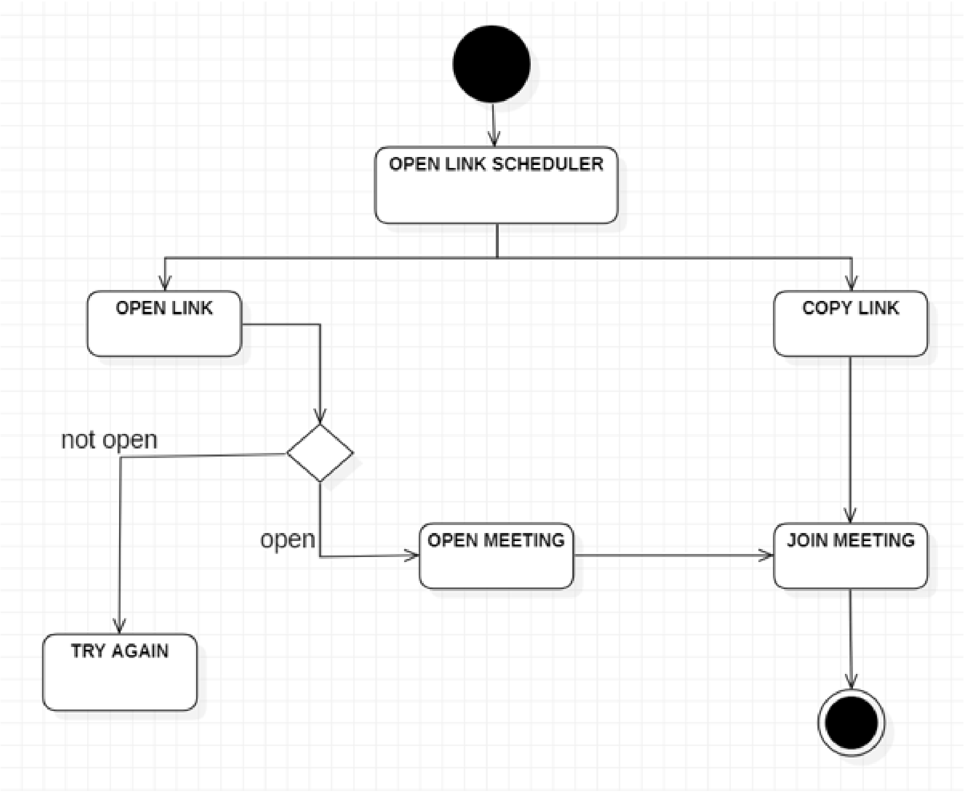
Use Case Diagram With Description

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analyzed to gather its functionalities, use cases are prepared and actors are identified.

Currently, LINK SCHEDULER is being used by the entire organization. The sales department has been using LINK SCHEDULER to communicate with clients.



Activity diagram –



Result:

Thus, the system architecture, use case and class diagram created successfully.

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	7
Title of Experiment	Design a Entity relationship diagram
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA, BHUVANESH E S, PRANEETH NARRA , K CHANDRA KIRAN
Register Number	RA2011003010015, RA2011003010021, RA2011003010012, RA2011003010020
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

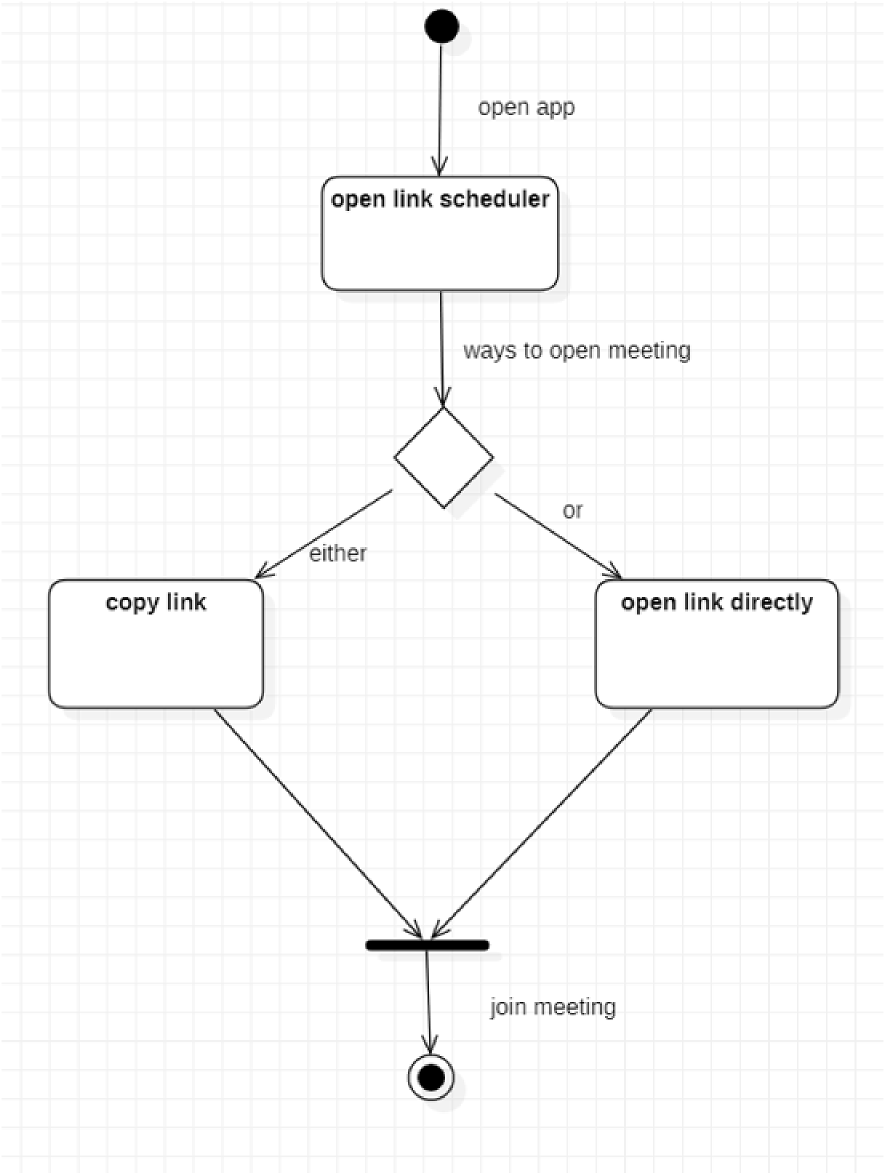
To create the Entity Relationship Diagram

Team Members:

S No	Register No	Name	Role
1	RA2011003010015	AKKASH ANUMALA	Rep
2	RA2011003010021	BHUVANESH E S	Member
3	RA2011003010012	PRANEETH NARRA	Member
4	RA2011003010020	K CHANDRA KIRAN	Member

Entity Relationship Diagram:

An entity relationship diagram (ERD), also known as an entity relationship model, is a graphical representation that depicts relationships among people, objects, places, concepts or events within an information technology (IT) system.



Result:

Thus, the entity relationship diagram was created successfully.

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	8
Title of Experiment	Develop a Data Flow Diagram (Process-Up to Level 1)
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA, BHUVANESH E S, PRANEETH NARRA , K CHANDRA KIRAN
Register Number	RA2011003010015, RA2011003010021, RA2011003010012, RA2011003010020
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

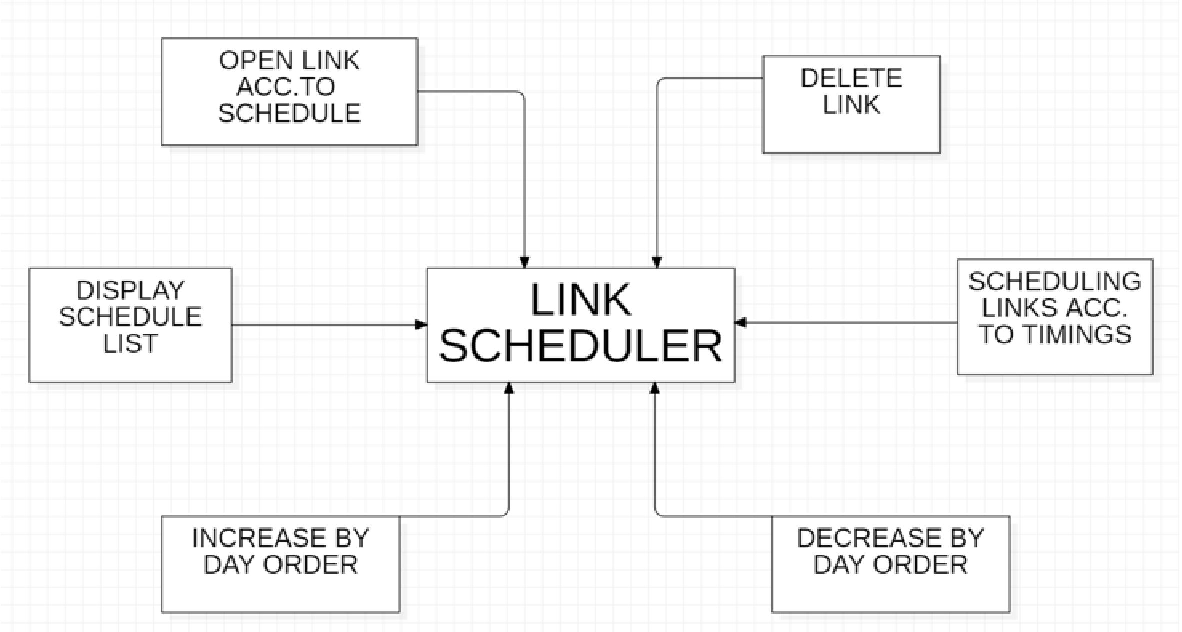
To develop the data flow diagram up to level 1 for the <project name>

Team Members:

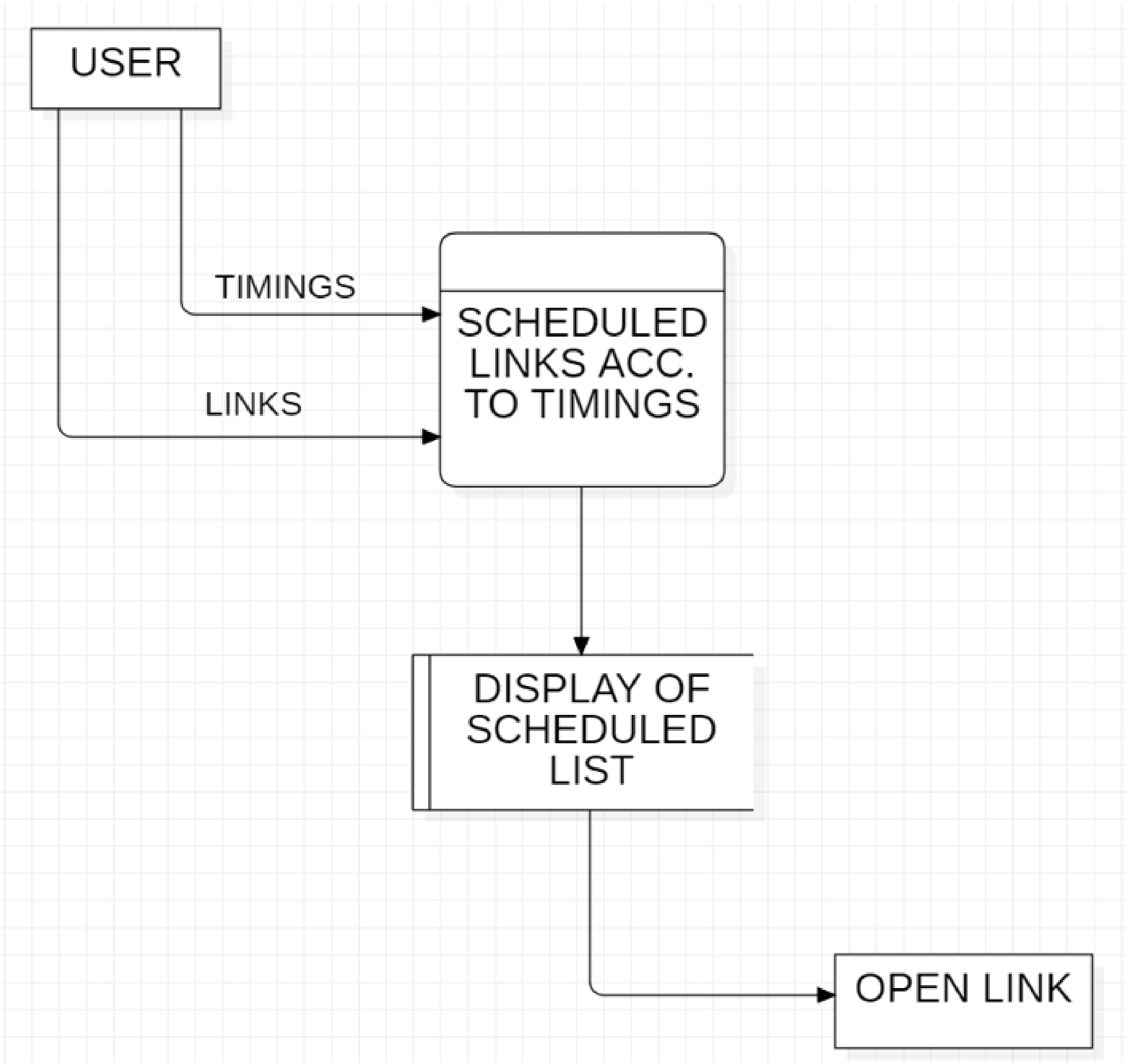
S No	Register No	Name	Role
1	RA2011003010015	AKKASH ANUMALA	Rep
2	RA2011003010021	BHUVANESH E S	Member
3	RA2011003010012	PRANEETH NARRA	Member
4	RA2011003010020	K CHANDRA KIRAN	Member

Data Flow Diagram :

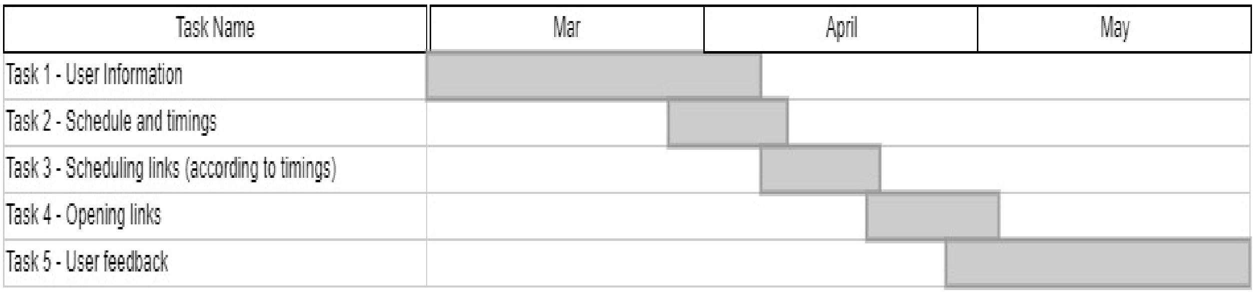
A data flow diagram (DFD) **maps out the flow of information for any process or system**. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination.



Upto Level 1 :



Gantt Chart :



Result:

Thus, the data flow diagrams have been created for the <project name>.

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	9
Title of Experiment	Design a Sequence and Collaboration Diagram
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA , BHUVANESH E S, PRANEETH NARRA, K CHANDRA KIRAN
Register Number	RA2011003010015, RA2011003010021, RA2011003010012, RA2011003010020
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

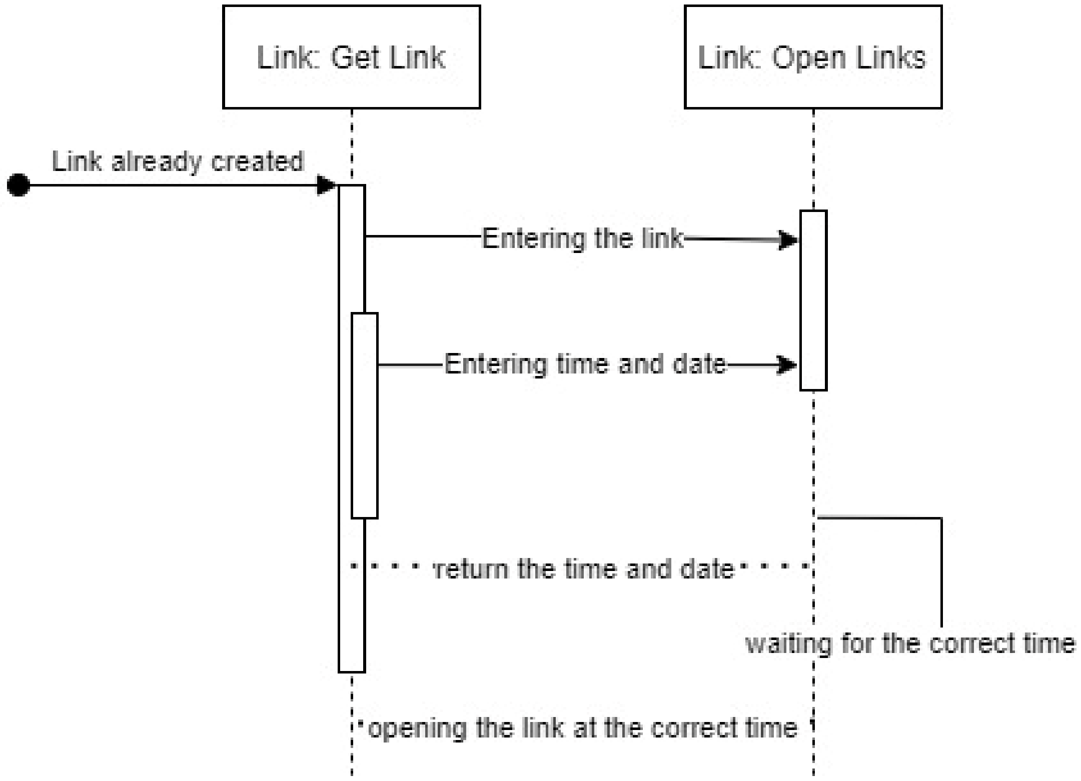
Aim

To create the sequence and collaboration diagram for the Link Scheduler

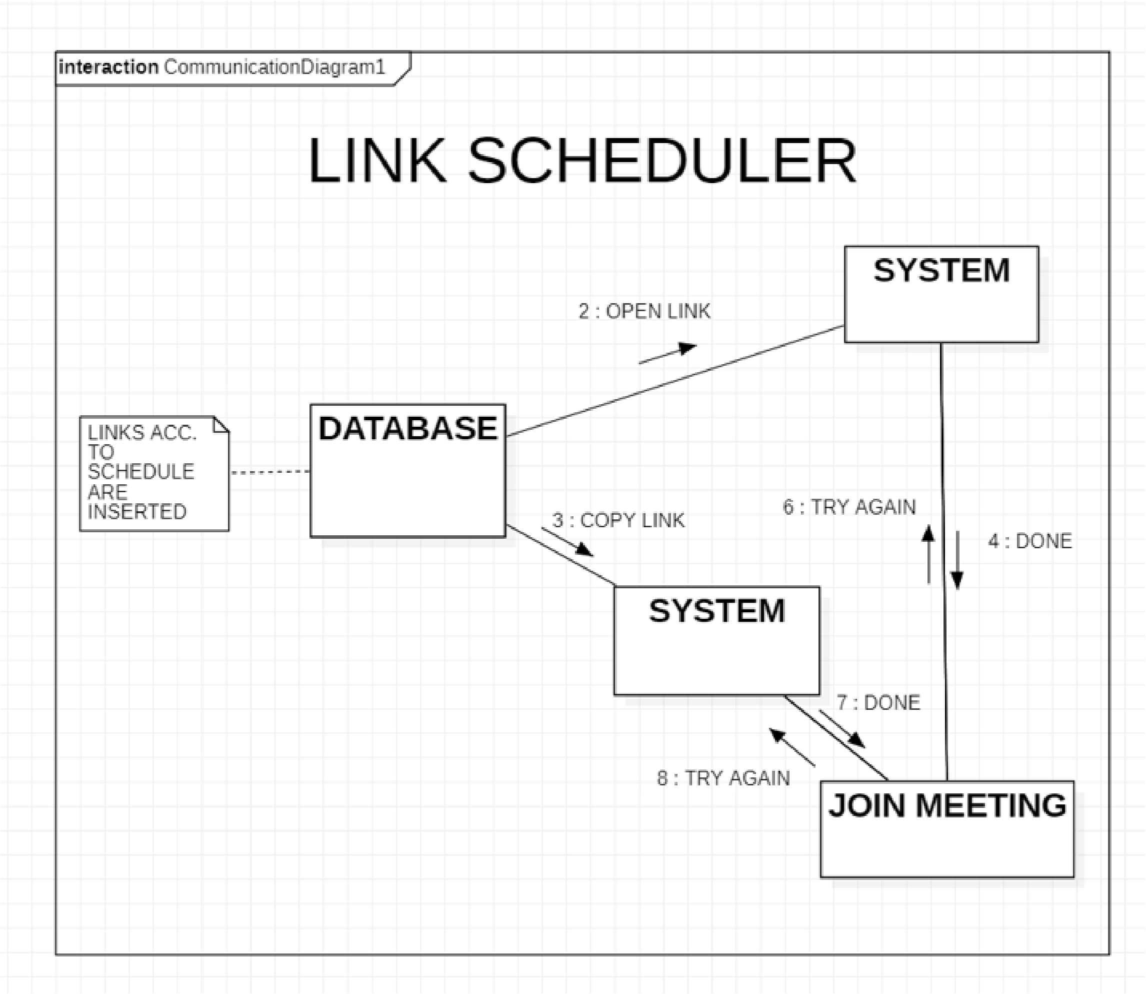
Team Members:

S No	Register No	Name	Role
1	RA2011003010015	AKKASH ANUMALA	Rep/Member
2	RA2011003010021	BHUVANESH E S	Member
3	RA2011003010012	PRANEETH NARRA	Member
4	RA2011003010020	K CHANDRA KIRAN	Member

Sequence Diagram:



Collaboration Diagram :



Result:

Thus, the sequence and collaboration diagrams were created for the <project name>.

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	10
Title of Experiment	Develop a Testing Framework/User Interface
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA, BHUVANESH E S, PRANEETH NARRA, K CHANDRA KIRAN
Register Number	RA2011003010015, RA2011003010021, RA2011003010012, RA2011003010020
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 develop the testing framework and/or user interface framework for the Link Scheduler

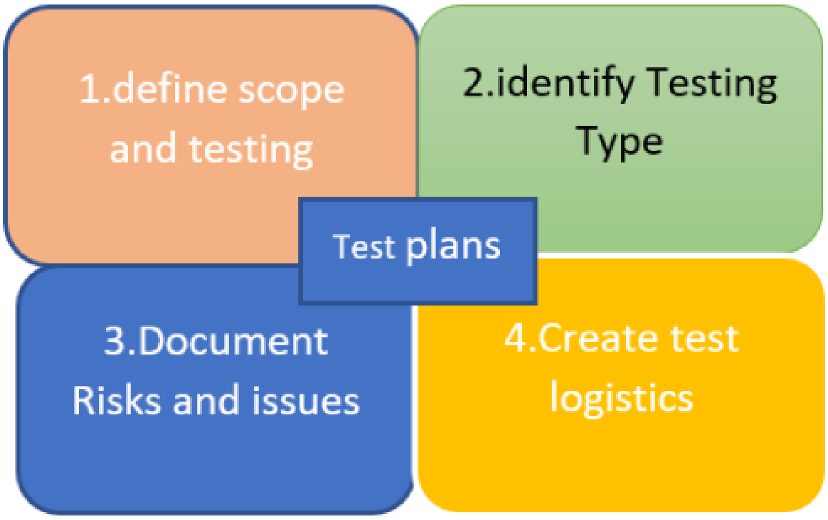
Team Members:

S No	Register No	Name	Role
1	AKKASH ANUMALA	RA2011003010015	Rep/Member
2	BHUVANESH E S	RA2011003010021	Member
3	PRANEETH NARRA	RA2011003010012	Member
4	K CHANDRA KIRAN	RA2011003010020	Member

Executive Summary

- To design effective test cases for identifying the errors
- To check whether all the calculations required in the application are correct
- Finding errors or defects which may be created while developing the Link Scheduler
- To be confident about the product.

Test Plan



Scope of Testing

Functional:

Check if the features and operational behavior of a product are as per specifications

Non-Functional:

NON-FUNCTIONAL TESTING is defined as a type of Software testing to check non-functional aspects (performance, usability, reliability, etc…) of a software application. It is designed to test the readiness of a system as per nonfunctional parameters which are never addressed by functional testing.

Types of Testing, Methodology, Tools

category	**Methodology**	**Tools required**
Functional requirements	**Unit Testing**	Unit test frameworks like c and C++ unit testing framework
Functional requirements	**Integration Testing**	Tools like git and github are required to merge.
Functional requirements	**System testing**	Testing the whole system using automation.
Non - Functional requirements	**Load testing**	Use tools in OS to see if the Application the loads.
Non - Functional requirements	**Stress Testing**	Use Github to share and simulate users and see if people can use it.

Result:

Thus, the testing framework/user interface framework has been created for the Link Scheduler.

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	11
Title of Experiment	Test Cases
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA , BHUVANESH E S, PRANEETH NARRA , K CHANDRA KIRAN
Register Number	RA2011003010015, RA2011003010021, RA2011003010012, RA2011003010020
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 develop the test cases manual for the Link Scheduler

Team Members:

S No	Register No	Name	Role
1	**AKKASH ANUMALA**	**RA2011003010015**	**Rep**
2	**BHUVANESH E S**	**RA2011003010021**	**Member**
3	**PRANEETH NARRA**	**RA2011003010012**	**Member**
4	**K CHANDRA KIRAN**	**RA2011003010020**	**Member**

Test Case

Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1	Check whether the application accepts the link entered by the user	Accept the link and saving it to the database	1.User types the appropriate number for entering the new link 2.Enter the link 3.Enter Date and time	It should again come to the home page and stores the link details in the timeline	It should again come to the home page and stores the link details in the timeline	Pass	success
2	Check if the entered link is available in the timeline	Go to the timeline and show the entered links	1.Type the appropriate number for the timeline 2.Go to the timeline	The link we entered should be in the timeline	The entered link is there in the timeline	Pass	Success
3	Check if we can delete the link we entered	Go to the delete links page to delete the link	1.Go to the delete link page 2.Enter the correct name of the link to delete it	The link should be deleted	The link was deleted successfully	pass	Success

Non-Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1	Links working	The link should open at the correct time	1.The entered link will be running in the background process 2.The background service should be turned on	The link should be opened on the correct time	The link opened 5 minutes before the meeting The expired links will not open	pass	Expired links will not work success
2	Review the defeats to the development team	Access ibility	The numbering is now used to open the pages in the application	The app will be modified with more user friendly features in the future	-----	Positive/ modified	Success

Result:

Thus, the test case manual has been created for the Link scheduler.

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	12
Title of Experiment	Manual Test Case Reporting
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA, BHUVANESH E S, PRANEETH NARRA , K CHANDRA KIRAN
Register Number	RA2011003010015, RA2011003010021, RA2011003010012, RA2011003010020
Date of Experiment	

Mark Split Up

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

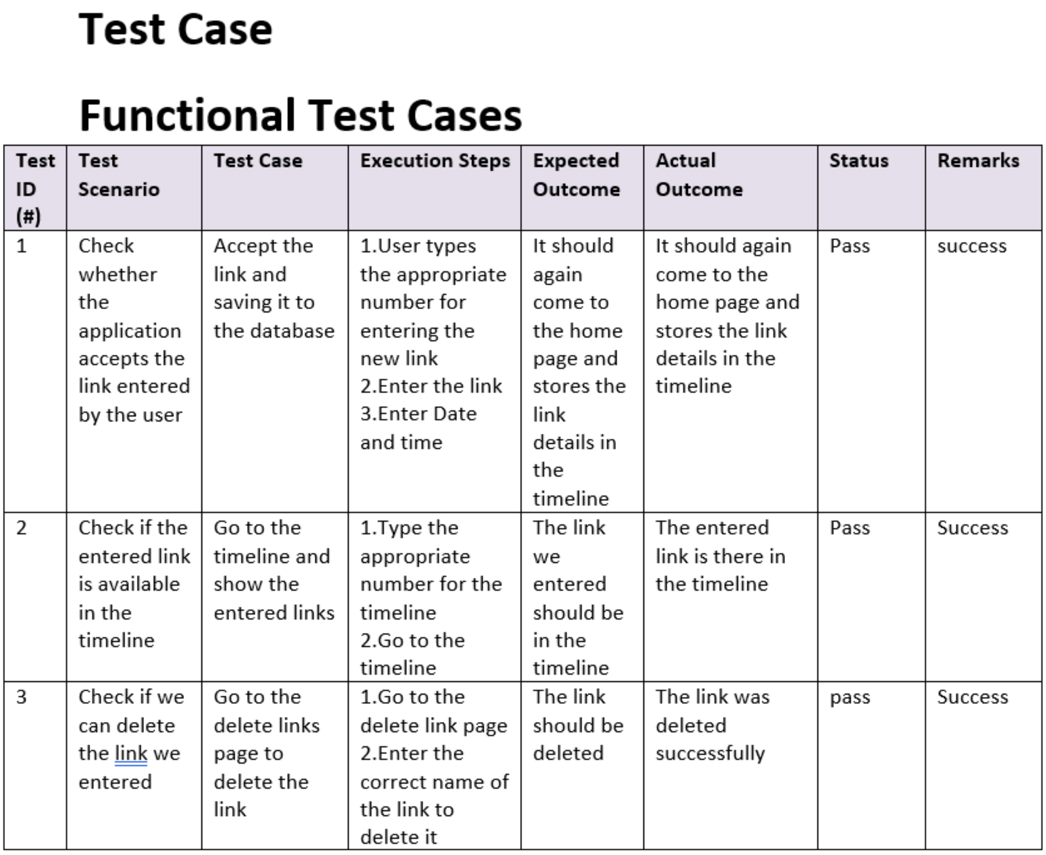
Staff Signature with date

Aim

To prepare the manual test case report for the Link scheduler

Team Members:

S No	Register No	Name	Role
1	AKKASH ANUMALA	RA2011003010015	Rep/Member
2	BHUVANESH E S	RA2011003010021	Member
3	PRANEETH NARRA	RA2011003010012	Member
4	K CHANDRA KIRAN	RA2011003010020	Member

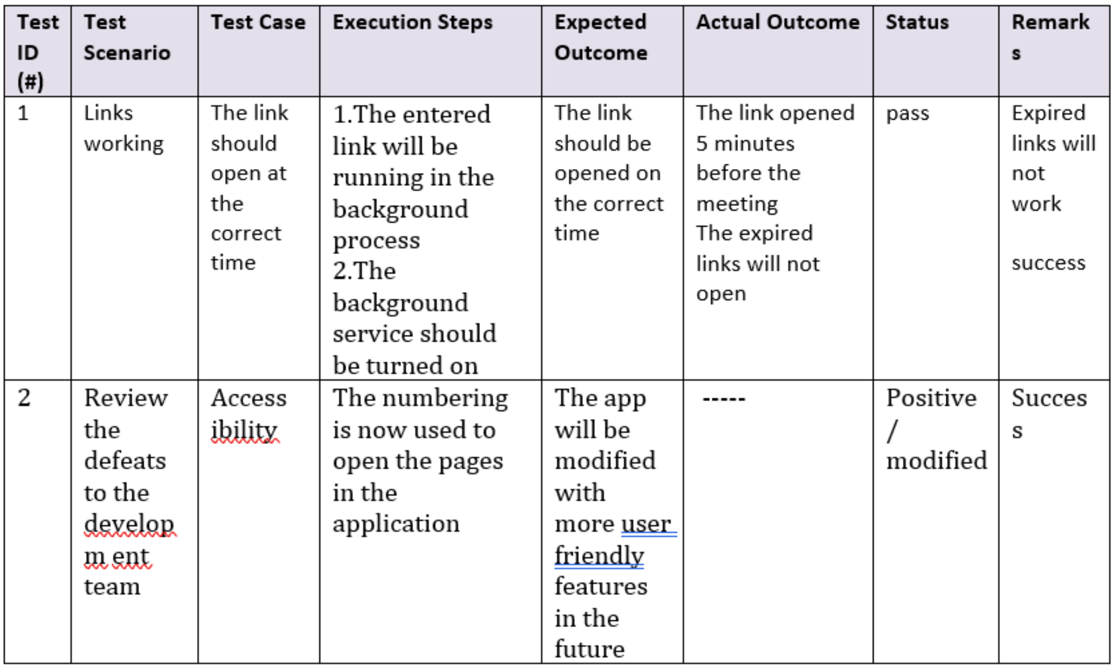


Test Case

Functional Test Cases

Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remarks
1	Check whether the application accepts the link entered by the user	Accept the link and saving it to the database	1.User types the appropriate number for entering the new link 2.Enter the link 3.Enter Date and time	It should again come to the home page and stores the link details in the timeline	It should again come to the home page and stores the link details in the timeline	Pass	success
2	Check if the entered link is available in the timeline	Go to the timeline and show the entered links	1.Type the appropriate number for the timeline 2.Go to the timeline	The link we entered should be in the timeline	The entered link is there in the timeline	Pass	Success
3	Check if we can delete the link we entered	Go to the delete links page to delete the link	1.Go to the delete link page 2.Enter the correct name of the link to delete it	The link should be deleted	The link was deleted successfully	pass	Success

Non-Functional Test Cases



Test ID (#)	Test Scenario	Test Case	Execution Steps	Expected Outcome	Actual Outcome	Status	Remark s
1	Links working	The link should open at the correct time	1.The entered link will be running in the background process 2.The background service should be turned on	The link should be opened on the correct time	The link opened 5 minutes before the meeting The expired links will not open	pass	Expired links will not work success
2	Review the defeats to the develop m ent team	Access ibility	The numbering is now used to open the pages in the application	The app will be modified with more user friendly features in the future	-----	Positive / modified	Succes s

Seek Help From Stakeholders To Remove Obstacles:

Seek stakeholders to understand the differences of opinions and make them transparent, carefully leading individuals and groups to find common ground. This is more than consensus.

Category	Progress Against Plan	Status
Functional Testing	Green	Completed
Non-Functional Testing	Green	Completed

Functional	Test Case Coverage (%)	Status
Start page	20%	Completed
Timeline Page	40%	Completed
Delete link page	40%	Completed

Result:

Thus, the test case report has been created for the Link scheduler

SRM INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be University u/s 3 of UGC Act, 1956)

School of Computing

SRM IST, Kattankulathur – 603 203

Course Code: 18CSC206J

Course Name: Software Engineering and Project Management

Experiment No	13
Title of Experiment	Provide the details of Architecture Design/Framework/Implementation
Name of the candidate	AKKASH ANUMALA
Team Members	AKKASH ANUMALA, BHUVANESH E S, PRANEETH NARRA, K CHANDRA KIRAN
Register Numbers	RA2011003010015, RA2011003010021, RA2011003010012, RA2011003010020
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

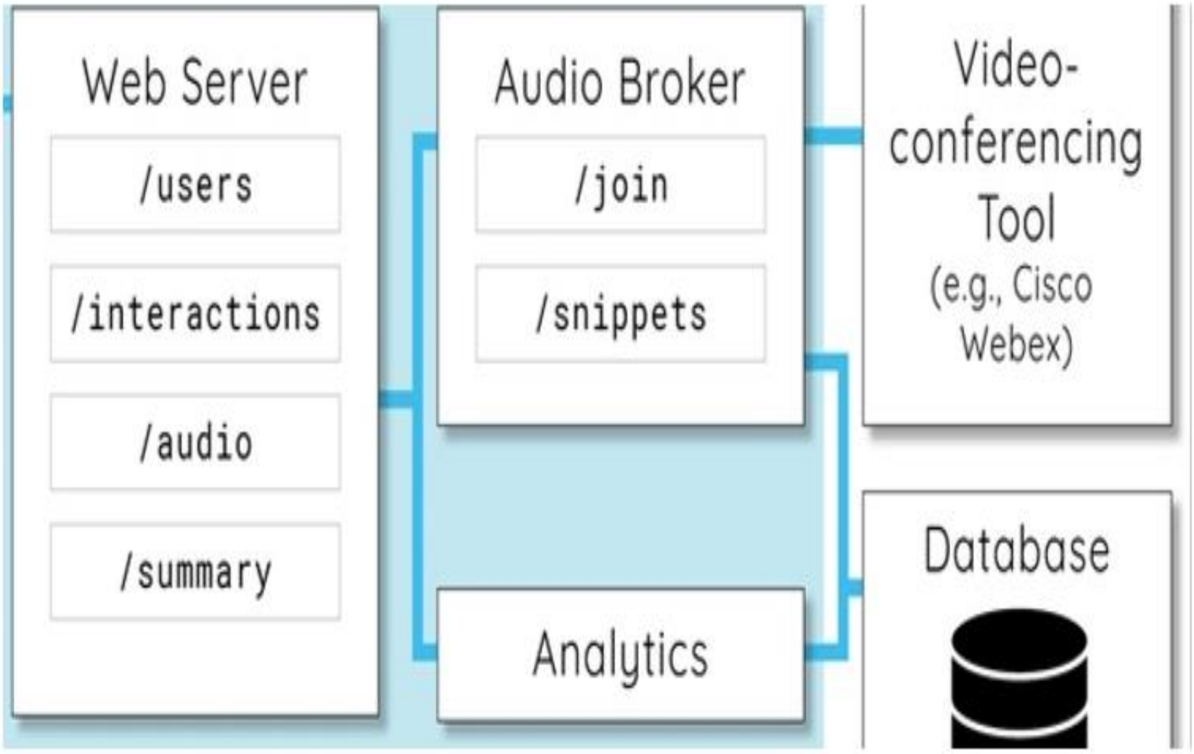
Aim

To provide the details of architectural design/framework/implementation

Team Members:

S No	Register No	Name	Role
1	RA2011003010015	AKKASH ANUMALA	Rep/Member
2	RA2011003010021	BHUVANESH E S	Member
3	RA2011003010012	PRANEETH NARRA	Member
4	RA2011003010020	K CHANDRA KIRAN	Member

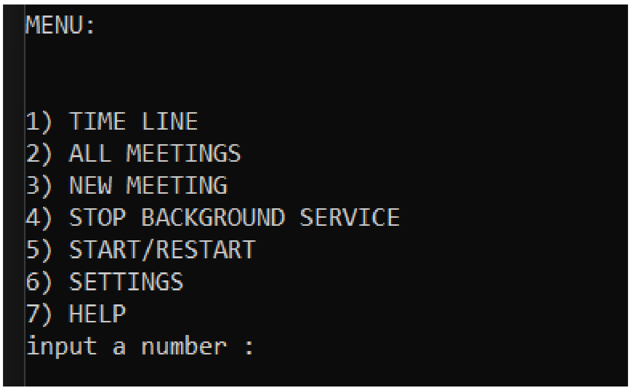
System architecture



IMPLEMENTATION :

Screenshots of Application

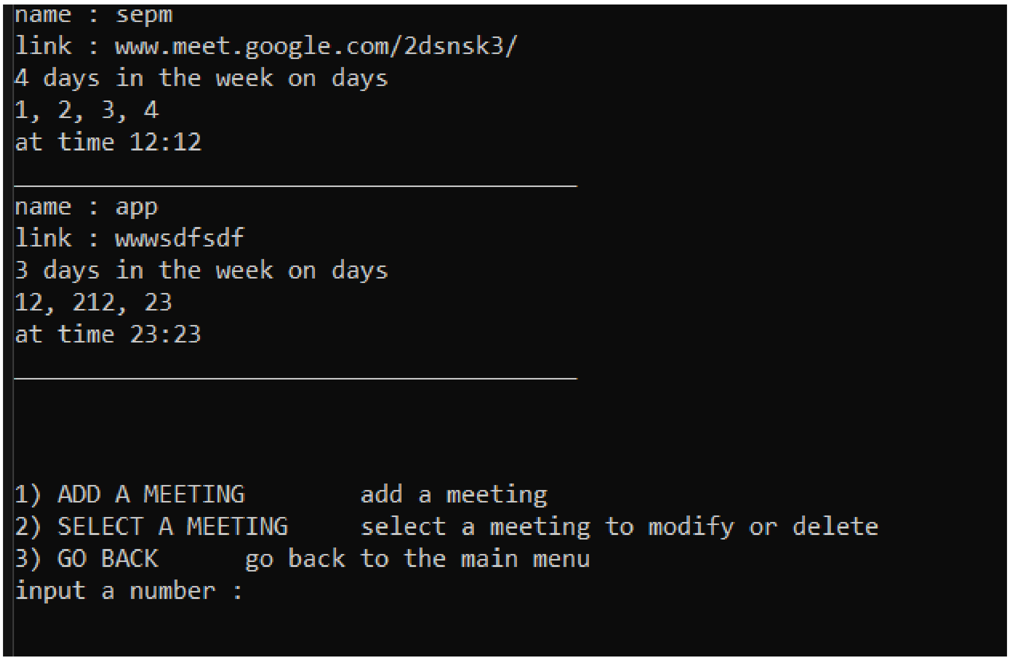
Home Screen



```
MENU:


1) TIME LINE
2) ALL MEETINGS
3) NEW MEETING
4) STOP BACKGROUND SERVICE
5) START/RESTART
6) SETTINGS
7) HELP
input a number :
```

All meetings screen

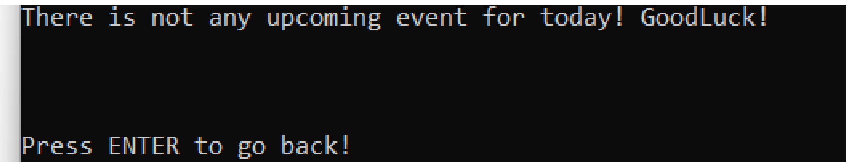


```
name : sepm
link : www.meet.google.com/2dsnsk3/
4 days in the week on days
1, 2, 3, 4
at time 12:12
_______________________________________
name : app
link : wwwsdfsdf
3 days in the week on days
12, 212, 23
at time 23:23
_______________________________________



1) ADD A MEETING        add a meeting
2) SELECT A MEETING     select a meeting to modify or delete
3) GO BACK      go back to the main menu
input a number :
```

Timeline screen

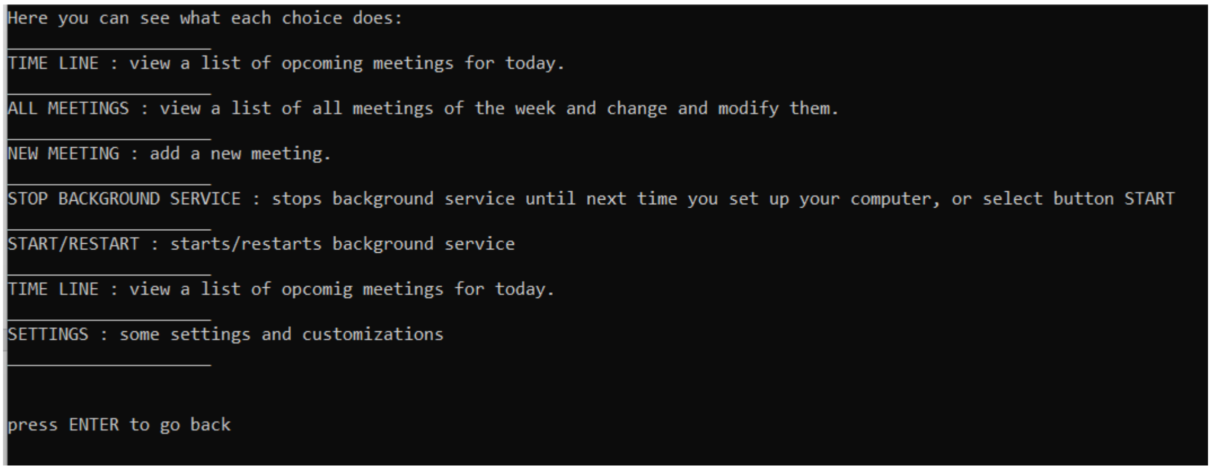


```
There is not any upcoming event for today! GoodLuck!



Press ENTER to go back!
```

Help screen



```
Here you can see what each choice does:
____________________
TIME LINE : view a list of opcoming meetings for today.
____________________
ALL MEETINGS : view a list of all meetings of the week and change and modify them.
____________________
NEW MEETING : add a new meeting.
____________________
STOP BACKGROUND SERVICE : stops background service until next time you set up your computer, or select button START
____________________
START/RESTART : starts/restarts background service
____________________
TIME LINE : view a list of opcomig meetings for today.
____________________
SETTINGS : some settings and customizations
____________________


press ENTER to go back
```

Code :

For the main Application :

```
#include <iostream>
#include <sstream>
#include <fstream>
#include <stdlib.h>
#include <ctime>
#include <time.h>
#include <windows.h>
#include <windowsx.h>
#include <cstdlib>
#include <stdio.h>
#include <string.h>



using namespace std;

int button_menu(string text, int number_of_choices, string choices[]);
void time_line_menu();
void all_meetings_menu();
void add_new_meeting_menu();
void resave_data();
void select_and_modify_menu();
void help();
void settings();
void initialize_settings();
void set_theme(int choice);

struct meeting {
    string name;
    string link;
    int number_of_days_in_week;
    int days_of_the_week[7];
    int hour;
    int minut;
};


int is_before(meeting meet1, meeting meet2) {
    /* returns 1 if meet1 be before meet2 */
    if (meet1.hour < meet2.hour || (meet1.hour == meet2.hour && meet1.minut <= meet2.minut)) {
        return 1;
    }
    return 0;
}

int get_number_of_day(const char* day_in_letters){
    char first_char = *day_in_letters;
    char second_char = *(day_in_letters+1);
    switch (first_char) {
    case 'S':
        if (second_char == 'u'){
            return 1;
        }
        if (second_char == 'a') {
            return 0;
        }
        break;
    case 'M':
        return 2;
        break;
    case 'T':
        if (second_char == 'u'){
            return 3;
        }
        if (second_char == 'h') {
            return 6;
        }
        break;
    case 'W':
        return 5;
    case 'F':
        return 7;
    default:
        return -1;
    }
    return -1;

}

meeting all_meetings[31];
int number_of_meetings = 0;
int exists_test() {
    if (FILE *file = fopen("C:/ProgramData/TeymooriAnar/AutoMeet/ListOfMeetings.txt", "r")) {
        fclose(file);
        return 1;
    } else {
        return 0;
    }
}
int exists_test_of_settings() {
    if (FILE *file = fopen("C:/ProgramData/user/AutoMeet/Settings.txt", "r")) {
        fclose(file);
        return 1;
    } else {
        return 0;
    }
}
void initialize_settings() {
    int exist = exists_test_of_settings();
    if (exist == 0) {
        system("mkdir \"C:/ProgramData/user/AutoMeet/\"");
        ofstream data_file("C:/ProgramData/user/AutoMeet/Settings.txt");
        data_file << "0";
        data_file.close();
    }

    string text_of_file = "";
    ifstream data_file("C:/ProgramData/user/AutoMeet/Settings.txt");
    getline(data_file, text_of_file);
    int number_of_theme;
    //stringstream number_of_theme_str;
    stringstream number_of_theme_str(text_of_file);
    number_of_theme_str >> number_of_theme;
    set_theme(number_of_theme);



}
void initialize_all_meetings() {
    int exist = exists_test();
    if (exist == 0) {
        system("mkdir \"C:/ProgramData/user/AutoMeet/\"");
        ofstream data_file("C:/ProgramData/user/AutoMeet/ListOfMeetings.txt");
        data_file << "";
        data_file.close();
    }

    string text_of_file;
    int number_of_meeting_to_save = 0;
    int number_of_days;
    stringstream number_of_days_str;
    int day;
    ifstream data_file("C:/ProgramData/user/AutoMeet/ListOfMeetings.txt");
    while (getline(data_file, text_of_file)) {
         // Output the text from the file
        meeting meet;
        meet.name = text_of_file;
        getline(data_file, text_of_file);
        meet.link = text_of_file;
        getline(data_file, text_of_file);
        stringstream number_of_days_str(text_of_file);
        number_of_days_str >> number_of_days;
        meet.number_of_days_in_week = number_of_days;
        while (number_of_days > 0) {
            getline(data_file, text_of_file);
            stringstream number_of_days_str(text_of_file);
            number_of_days_str >> day;
            meet.days_of_the_week[meet.number_of_days_in_week-number_of_days] = day;
            number_of_days --;
        }
        getline(data_file, text_of_file);
        stringstream number_of_hour_str(text_of_file);
        number_of_hour_str >> day;
        meet.hour = day;

        getline(data_file, text_of_file);
        stringstream number_of_min_str(text_of_file);
        number_of_min_str >> day;
        meet.minut = day;
        all_meetings[number_of_meetings] = meet;
        number_of_meetings ++;

    }
    // // Close the file
     data_file.close();
}
```

```
int is_for_today(meeting meet, int today_int) {
    for (int i=0; i<meet.number_of_days_in_week; i++) {
        if (meet.days_of_the_week[i] == today_int) {
            return 1;
        }
    }
    return 0;
}

int main() {
    initialize_all_meetings();
    initialize_settings();

    string choices[] = {"TIME LINE",
                "ALL MEETINGS",
                "NEW MEETING",
                "STOP BACKGROUND SERVICE",
                "START/RESTART",
                "SETTINGS",
                "HELP"};

    while(1) {
        system("cls");
        int choice = button_menu("Hello! Welcome to Link Scheduler app !\n\nMENU:\n\n",
                  7,
                  choices);
        switch (choice) {
            case 1:
                time_line_menu();
                break;
            case 2:
                all_meetings_menu();
                break;
            case 3:
                add_new_meeting_menu();
                break;
            case 4:
                system("taskkill /im AutoMeet(BackgroundService).exe");
                break;
            case 5:
                system("taskkill /im AutoMeet(BackgroundService).exe");
                system("start AutoMeet(BackgroundService).exe");
                break;
            case 6:
                settings();
                break;
            case 7:
                help();
                break;
        }
    }
}

void set_theme(int choice) {
    int all_colors = 17;
    choice = (choice % all_colors) + 1;
    switch (choice) {
    case 1:
        system("color 07");
        break;
    case 2:
        system("color 70");
        break;
    case 3:
        system("color 87");
        break;
    case 4:
        system("color 78");
        break;
    case 5:
        system("color 24");
        break;
    case 6:
        system("color 42");
        break;
    case 7:
        system("color 82");
        break;
    case 8:
        system("color 28");
        break;
    case 9:
        system("color 34");
        break;
    case 10:
        system("color 43");
        break;
    case 11:
        system("color 35");
        break;
    case 12:
        system("color 53");
        break;
    case 13:
        system("color 5c");
        break;
    case 14:
        system("color 31");
        break;
    case 15:
        system("color 13");
        break;
    case 16:
        system("color 1a");
        break;
    case 17:
        system("color a1");
```

```
        break;
    }
    ofstream data_file("C:/ProgramData/user/AutoMeet/Settings.txt");
    data_file << "" << (choice-1);
    data_file.close();
}

void theme_select() {
    int color_to_set;
    cout << "enter an integer" << endl;
    fflush(stdin);
    scanf("%d", &color_to_set);
    set_theme(color_to_set);
    return;
}

void settings() {
    string choices[] = {"THEME", "BACK"};
    while (1) {
        system("cls");
        int choice = button_menu("========SETTINGS========",
                    2,
                    choices);
        if (choice == 2) {
            break;
        } else {
            theme_select();
        }
    }
    return;
}

void help() {
    cout << "Here you can see what each choice does:" << endl;
    cout << "______________________" << endl;
    cout << "TIME LINE : view a list of opcoming meetings for today." << endl;
    cout << "____________________" << endl;
    cout << "ALL MEETINGS : view a list of all meetings of the week and change and modify them." << endl;
    cout << "____________________" << endl;
    cout << "NEW MEETING : add a new meeting." << endl;
    cout << "____________________" << endl;
    cout << "STOP BACKGROUND SERVICE : stops background service until next time you set up your computer, or select button
START" << endl;
    cout << "____________________" << endl;
    cout << "START/RESTART : starts/restarts background service" << endl;
    cout << "____________________" << endl;
    cout << "TIME LINE : view a list of opcomig meetings for today." << endl;
    cout << "____________________" << endl;
    cout << "SETTINGS : some settings and customizations" << endl;
    cout << "____________________" << endl;
    cout << "\n\npress ENTER to go back" << endl;
    fflush(stdin);
    scanf("%*c");
    return;
}

int button_menu(string text, int number_of_choices, string choices[]) {
    cout << text << endl;
    for (int i = 0; i < number_of_choices; i++)
    {
        cout << (i+1) << ") " << choices[i] << endl;
    }
    int choice;
    string choice_string;
    do {
        cout << "input a number : ";
        cin >> choice_string;
        stringstream choice_parser(choice_string);
        choice_parser >> choice;
        if (choice > number_of_choices || choice <= 0)
        {
            cout << "You should enter an integer between 1 and " << number_of_choices << ".\nTry again!\n" <<endl;
        }

    } while (choice <= 0 || choice > number_of_choices);
    system("cls");
    return choice;
}

void all_meetings_menu() {
    for (int i = 0; i < number_of_meetings; i++) {
        cout << "name : " << all_meetings[i].name << "\nlink : " << all_meetings[i].link << "\n" <<
all_meetings[i].number_of_days_in_week
                << (all_meetings[i].number_of_days_in_week == 1 ? " day in the week on day \n" : " days in the week on days \n");
        for (int j = 0; j < all_meetings[i].number_of_days_in_week; j++)
        {
            cout << all_meetings[i].days_of_the_week[j] << (j != all_meetings[i].number_of_days_in_week-1 ? ", " : "\n");
        }
        cout << "at time " << all_meetings[i].hour << ":" << all_meetings[i].minut << "\n________________________________________"
<< endl;

    }
    string choices[] = {"ADD A MEETING\tadd a meeting",
                "SELECT A MEETING\tselect a meeting to modify or delete",
                "GO BACK\tgo back to the main menu"};
    int button_choice = button_menu("\n\n", 3, choices);
    switch (button_choice){
        case 1:
            add_new_meeting_menu();
            break;
        case 2:
            select_and_modify_menu();
            break;
        case 3:
            return;
        default:
            break;
    }
```

```
}

void time_line_menu() {
    time_t now;
    char* dt;
    int today_number;
    meeting meet;
    int opened;
    int is_any_upcoming;
    int next_meet_set = 0;
    meeting next_meet;
    int to_wait;
    // current date/time based on current system
    now = time(0);
    opened = 0;
    // convert now to string form
    dt = ctime(&now);
    tm *ltm = localtime(&now);
    today_number = get_number_of_day(dt);
    is_any_upcoming = 0;
    for (int i = 0; i < number_of_meetings; i++) {
        meet = all_meetings[i];
        if (is_for_today(meet, today_number) == 1) {

            if (meet.hour > ltm->tm_hour || (meet.hour == ltm->tm_hour && meet.minut > ltm->tm_min)){
                cout << "There is an upcoming meeting for today : " << meet.name << "\tat time\t" << meet.hour << ":" << meet.minut <<
endl;
                if(next_meet_set == 0) {
                    next_meet = meet;
                    next_meet_set = 1;
                } else {
                    if (is_before(meet, next_meet)) {
                        next_meet = meet;
                    }
                }
                is_any_upcoming = 1;
            }
            if ((ltm->tm_hour == meet.hour && ltm->tm_min <= meet.minut && ltm->tm_min >= meet.minut - 30)||(ltm->tm_hour ==
meet.hour - 1 && ltm->tm_min-30>=meet.minut)) {
                string command = "start "+meet.link;
                const char * command_char_ptr = command.c_str();
                cout << "Opening meeting "<<meet.name<<" for time "<<meet.hour<<":"<<meet.minut<<" ... "<<endl;
                system(command_char_ptr);
                opened = 1;
            }
        }

    }
    if (is_any_upcoming == 0) {
        cout << "There is not any upcoming event for today! GoodLuck!" << endl;
    }
    cout << "\n\n\nPress ENTER to go back!" << endl;
    fflush(stdin);
    getchar();
    system("cls");
    return;

}

void add_new_meeting_menu() {
    meeting new_meet;
    system("cls");
    cout << "NEW MEETING" << endl;
    if (number_of_meetings >= 30) {
        cout << "You can at most have 30 meetings.\nEach meeting can be repeated at most 7 days a week." << endl;
        cout << "Delete at least meeting to be able to add new meetings;)" << endl;
        return;
    }
    cout << "To cancel, just close the application window!" << endl;
    cout << "\n\nStep1 : Select a name for the meeting.\n\nname : ";
    fflush(stdin);
    getline(cin, new_meet.name);
    system("cls");
    fflush(stdin);
    cout << "NEW MEETING|" << new_meet.name << endl;
    cout << "To cancel, just close the application window!" << endl;
    cout << "\n\nStep2 : Enter the link of meeting to open when the meeting starts\n\nlink : ";
    cin >> new_meet.link;
    system("cls");
    fflush(stdin);
    cout << "NEW MEETING|" << new_meet.name << endl;
    cout << "To cancel, just close the application window!" << endl;
    cout << "\n\nStep3 : Enter the number of meeting's days in the week\n\nnumber of days in a week : ";
    string number_of_days_string;
    cin >> number_of_days_string;
    stringstream number_Of_days_int(number_of_days_string);
    number_Of_days_int >> new_meet.number_of_days_in_week;
    system("cls");
    fflush(stdin);
    cout << "NEW MEETING|" << new_meet.name << endl;
    cout << "To cancel, just close the application window!" << endl;
    cout << "\n\nStep4 : Enter the meeting's days in the week line by line\n0 Means Saturday...., and 7 means Friday\n\nnumber of
day in a week : " << endl;
    for (int i = 0; i < new_meet.number_of_days_in_week; i++)
    {

        fflush(stdin);
        cin >> number_of_days_string;
        stringstream number_Of_days_int(number_of_days_string);
        number_Of_days_int >> new_meet.days_of_the_week[i];
    }

    system("cls");
    fflush(stdin);
    cout << "NEW MEETING|" << new_meet.name << endl;
    cout << "To cancel, just close the application window!" << endl;
    cout << "\n\nStep5 : Enter the meeting's time\n\nhour : ";
    cin >> number_of_days_string;
    stringstream hour_int(number_of_days_string);
    hour_int >> new_meet.hour;
    cout << "minute : ";
    fflush(stdin);
    cin >> number_of_days_string;
    stringstream minut_int(number_of_days_string);
        minut_int >> new_meet.minut;

    all_meetings[number_of_meetings] = new_meet;
    number_of_meetings ++;

    resave_data();


}

void resave_data() {
    string text_of_file_to_save = "";
    int number;
    string str;
    for (int i = 0; i < number_of_meetings; i++)
    {
        stringstream ss;
        ss << all_meetings[i].number_of_days_in_week;
        ss >> str;
        text_of_file_to_save += all_meetings[i].name+"\n"+all_meetings[i].link+"\n"+str+"\n";
        for (int j = 0; j < all_meetings[i].number_of_days_in_week; j++)
        {
            stringstream ssday;
            ssday << all_meetings[i].days_of_the_week[j];
            ssday >> str;
            text_of_file_to_save += str+"\n";
        }

        stringstream sshour;
        sshour << all_meetings[i].hour;
        sshour >> str;
        text_of_file_to_save += str+"\n";
        stringstream ssmin;
        ssmin << all_meetings[i].minut;
        ssmin >> str;
        text_of_file_to_save += str;
        if (i < number_of_meetings - 1)
        {
            text_of_file_to_save += "\n";
        }

    }

    ofstream data_file("C:/ProgramData/user/AutoMeet/ListOfMeetings.txt");
    data_file << text_of_file_to_save;
    data_file.close();
    system("taskkill /im AutoMeet(BackgroundService).exe");
    system("start AutoMeet(BackgroundService).exe");

}

void select_and_modify_menu() {
    system("cls");
    string names[20];
    for (int i = 0; i < number_of_meetings; i++)
    {
        names[i] = all_meetings[i].name;
    }
    int choice = button_menu("Here is a list of names of all meetings. Decide one of them :\n", number_of_meetings, names);
    choice--;
    system("cls");
    cout << all_meetings[choice].name << endl;
    string options[] = {"OPEN NOW",
                        "CHANGE NAME",
                        "CHANGE LINK",
                "CHANGE HOUR",
                "CHANGE MINUT",
                "DELETE",
                "CANCEL"};
int choice2 = button_menu("What do you want to do?", 7, options);
if (choice2 == 1) {
  string command = "start "+all_meetings[choice].link;
        const char * command_char_ptr = command.c_str();
        cout << "Opening meeting "<<all_meetings[choice].name<<" ... "<<endl;
        system(command_char_ptr);
} else if (choice2 == 2)
{
  fflush(stdin);
  cout << "new name : ";
  string new_name;
  getline(cin, new_name);
  all_meetings[choice].name = new_name;
} else if (choice2 == 3) {
  fflush(stdin);
  cout << "new link : ";
  string new_link;
  getline(cin, new_link);
  all_meetings[choice].link = new_link;

} else if (choice2 == 4) {
  fflush(stdin);
  cout << "new hour : ";
  string new_hour_string;
  getline(cin, new_hour_string);
  stringstream ss(new_hour_string);
  int new_hour_int;
  ss >> new_hour_int;
  all_meetings[choice].hour = new_hour_int;
} else if (choice2 == 5) {
  fflush(stdin);
  cout << "new minute : ";
```

```
        string new_hour_string;
        getline(cin, new_hour_string);
        stringstream ss(new_hour_string);
        int new_hour_int;
        ss >> new_hour_int;
        all_meetings[choice].hour = new_hour_int;
    } else if (choice2 == 6) {
        for (int i = choice; i < number_of_meetings-1; i++)
        {
            all_meetings[i] = all_meetings[i+1];
        }
        number_of_meetings--;
    } else if (choice2 == 7) {
        return;
    }

    resave_data();
    return;
}
```

For the Background service :

```
#include <iostream>
#include <sstream>
#include <fstream>
#include <stdlib.h>
#include <ctime>
#include <time.h>
#include <windowsx.h>
#include <cstdlib>
#include <stdio.h>
#define _WIN32_WINNT 0x0500
#include <windows.h>
#include <WinUser.h>

using namespace std;
```

```
struct meeting {
    string name;
    string link;
    int number_of_days_in_week;
    int days_of_the_week[7];
    int hour;
    int minut;
};


int is_before(meeting meet1, meeting meet2) {
    if (meet1.hour < meet2.hour || (meet1.hour == meet2.hour && meet1.minut <= meet2.minut)) {
        return 1;
    }
    return 0;
}

int get_number_of_day(const char* day_in_letters){
    char first_char = *day_in_letters;
    char second_char = *(day_in_letters+1);
    switch (first_char) {
    case 'S':
        if (second_char == 'u'){
            return 1;
        }
        if (second_char == 'a') {
            return 0;
        }
        break;
    case 'M':
        return 2;
        break;
    case 'T':
        if (second_char == 'u'){
            return 3;
        }
        if (second_char == 'h') {
            return 6;
        }
        break;
    case 'W':
        return 5;
    case 'F':
        return 7;
    default:
        return -1;
    }
    return -1;

}

meeting all_meetings[31];
int number_of_meetings = 0;

void initialize_all_meetings() {
    string text_of_file;
    int number_of_meeting_to_save = 0;
    int number_of_days;
    stringstream number_of_days_str;
    int day;
    ifstream data_file("C:/ProgramData/user/AutoMeet/ListOfMeetings.txt");
    while (getline(data_file, text_of_file)) {
        meeting meet;
        meet.name = text_of_file;
        getline(data_file, text_of_file);
        meet.link = text_of_file;
        getline(data_file, text_of_file);
        stringstream number_of_days_str(text_of_file);
        number_of_days_str >> number_of_days;
        meet.number_of_days_in_week = number_of_days;
        while (number_of_days > 0) {
            getline(data_file, text_of_file);
            stringstream number_of_days_str(text_of_file);
            number_of_days_str >> day;
            meet.days_of_the_week[meet.number_of_days_in_week-number_of_days] = day;
            number_of_days --;
        }
        getline(data_file, text_of_file);
        stringstream number_of_hour_str(text_of_file);
        number_of_hour_str >> day;
        meet.hour = day;

        getline(data_file, text_of_file);
        stringstream number_of_min_str(text_of_file);
        number_of_min_str >> day;
        meet.minut = day;
        all_meetings[number_of_meetings] = meet;
        number_of_meetings ++;

    }
     data_file.close();
}

int is_for_today(meeting meet, int today_int) {
    for (int i=0; i<meet.number_of_days_in_week; i++) {
        if (meet.days_of_the_week[i] == today_int) {
            return 1;
        }
    }
    return 0;
}

int main() {
    system("title AutoMeet(BackgroundService)");
    initialize_all_meetings();

                ShowWindow( GetConsoleWindow(), SW_HIDE );
    time_t now;
    char* dt;
    int today_number;
    meeting meet;
    int opened;
    int is_any_upcoming;
    int next_meet_set = 0;
    meeting next_meet;
    int to_wait;
    while (1) {
        now = time(0);
        opened = 0;

        dt = ctime(&now);
        tm *ltm = localtime(&now);

        today_number = get_number_of_day(dt);
        is_any_upcoming = 0;
        for (int i = 0; i < number_of_meetings; i++) {
            meet = all_meetings[i];
            if (is_for_today(meet, today_number) == 1) {


                if (meet.hour > ltm->tm_hour || (meet.hour == ltm->tm_hour && meet.minut > ltm->tm_min)){
                    cout << "There is an upcoming meeting for today : " << meet.name << "\tat time\t" << meet.hour << ":" << meet.minut
<< endl;
                    if(next_meet_set == 0) {
                        next_meet = meet;
                        next_meet_set = 1;
                    } else {
                        if (is_before(meet, next_meet)) {
                            next_meet = meet;
                        }
                    }
                    is_any_upcoming = 1;
        }

        if ((ltm->tm_hour == meet.hour && ltm->tm_min <= meet.minut && ltm->tm_min >= meet.minut - 30)||(ltm->tm_hour ==
meet.hour - 1 && ltm->tm_min-30>=meet.minut)) {
          string command = "start "+meet.link;
          const char * command_char_ptr = command.c_str();
          cout << "Opening meeting "<<meet.name<<" for time "<<meet.hour<<":"<<meet.minut<<" ... "<<endl;
          system(command_char_ptr);
          Sleep(30*60000);
          opened = 1;
        }
      }

    }
    if (is_any_upcoming == 0) {
      cout << "There is not any upcoming meeting today! GoodLuck!" << endl;
    }

    cout << "Last time checked at: " << dt << endl;
    Sleep(180000);
    system("cls");
  }
```

Result:

Thus, the details of architectural design/framework/implementation along with the screenshots were provided.

Conclusion

This project aims to make your life a bit easier. In these times, when everything is being digital from studying from home to working from home. For that you have separate links for every meeting and managing those links is a bit difficult. Instead, this project will do that for you, all your meetings with specific links will be managed by this project for you and you will no longer need to use multiple apps for that.

References

www.geeksforgeeks.org

www.tutorialspoint.com

www.javatpoint.com

www.github.com