5408_RDBMS_CUSTOM (RDB)

DALHOUSIE UNIVERSITY

FACULTY OF COMPUTER SCIENCE

PROJECT PHASE 1

Feasibility Study

CSCI5408: Database Management, Warehousing, Analytics

Group Name: DP-4

Group Members

Dhruv Doshi	B00883311
Heenal Sapovadia	B00868350
Heli Niteenbhai Patel	B00868906

PROJECT OBJECTIVE

To build a simple custom relational database (RDb) and its management system (RDbMS)

TECHNOLOGIES AND FRAMEWORK

Apart from being one of the requirements of the project, there are many reasons of using Java for building our custom RDBMS system. Java has a Collections framework which offers a wide range of interfaces and classes which enable efficient storage and processing of data. Since data storage and processing is performed heavily in any database system, we can make use of the data structures under Collections framework to our benefit. A few examples of classes are ArrayList, LinkedList, Priority Queue and HashSet, and interfaces are Set, List and Queue.

LEARNINGS

All the group members are acquainted with Java as a programming language; hence coding would not be much of a challenge. Although we have started exploring the data structures required for the project, in depth understanding and implementation is something we will learn as we develop the project. Implementing the core components of the dbms like query optimization, transaction processor, locking mechanism sound challenging and will require sufficient brainstorming. We have explored the data structures in Collection's framework and have produced a tentative plan which is explained below.

DATA STRUCTURES

There are multiple ways to implement the given functionalities of the project using many different data structures, but the ones we plan to use are described below with reasoning.

HashMap

- To store the data being read from the persistent storage: HashMap support key-value paired data wherein the key is unique. This can be leveraged to store tables since table names of a database and the column names in a table must be unique.
- To perform In-memory operations: Since, HashMap support searching in O (1) time complexity, retrieving records from tables would become easier.

ArrayList and LinkedList

- To insert/delete/select data: ArrayList allows faster indexed access whereas LinkedList performs faster when it comes to addition and deletion of data. Hence, we are planning to use them as per the functionality we implement.

Stack and Queue

- To handle multiple queries and to work on complex query structures: Queue support FIFO (First in First out) mechanism hence we plan to use it for executing multiple queries. Stacks are based on LIFO (Last in First out) concept; hence they can be used for complex query structures like nested SQL statements.

HashSet

- To store the data of Primary Key columns: HashSet supports only unique values hence, it can be used to load the data of primary key columns and check for duplications while data insertion.

For query parsing purposed, we are planning to use the Pattern and Matcher class from Regex package provided in Java.

CUSTOM FILE FORMAT

Since standard file formats such as JSON, CSV and XML are not allowed for persistent storage, we are planning to use simple text(.txt) files with a pre decided format of records and other data.

METADATA (DATA DICTIONARY)

Metadata will be stored in text files as well. Currently, we plan to have the metadata for the following:

- Schema: Schema file will contain the details for all the tables in a database; columns, datatypes, constraints, foreign keys for each table. It will also contain the lower limit/upper limit of the datatypes.
- Security: Security file will contain information regarding the users, credentials, and privileges.

SQL DUMP

SQL dump file will be a basic text file generated custom file for loading data across platforms. For generation of this file, we will use the metadata stored in the text files to fetch the table structure and form CREATE statements and INSERT statements would be created using the data stored in data files.

ERD

We are planning to generate an ERD and store it in the form of a text file. The format in which we store the details of the database tables will be pre decided. The ERD will depict the table structure along with the relationships with other tables in the form of cardinality information.

INITIAL FLOWCHART FOR THE RELATIONAL DATABASE

According to the tentative plan we have developed flowchart as below (Figure -1). This flowchart consists of all stages connection, authentication, query validation and error messages, query processing which consists of query optimization and execution.

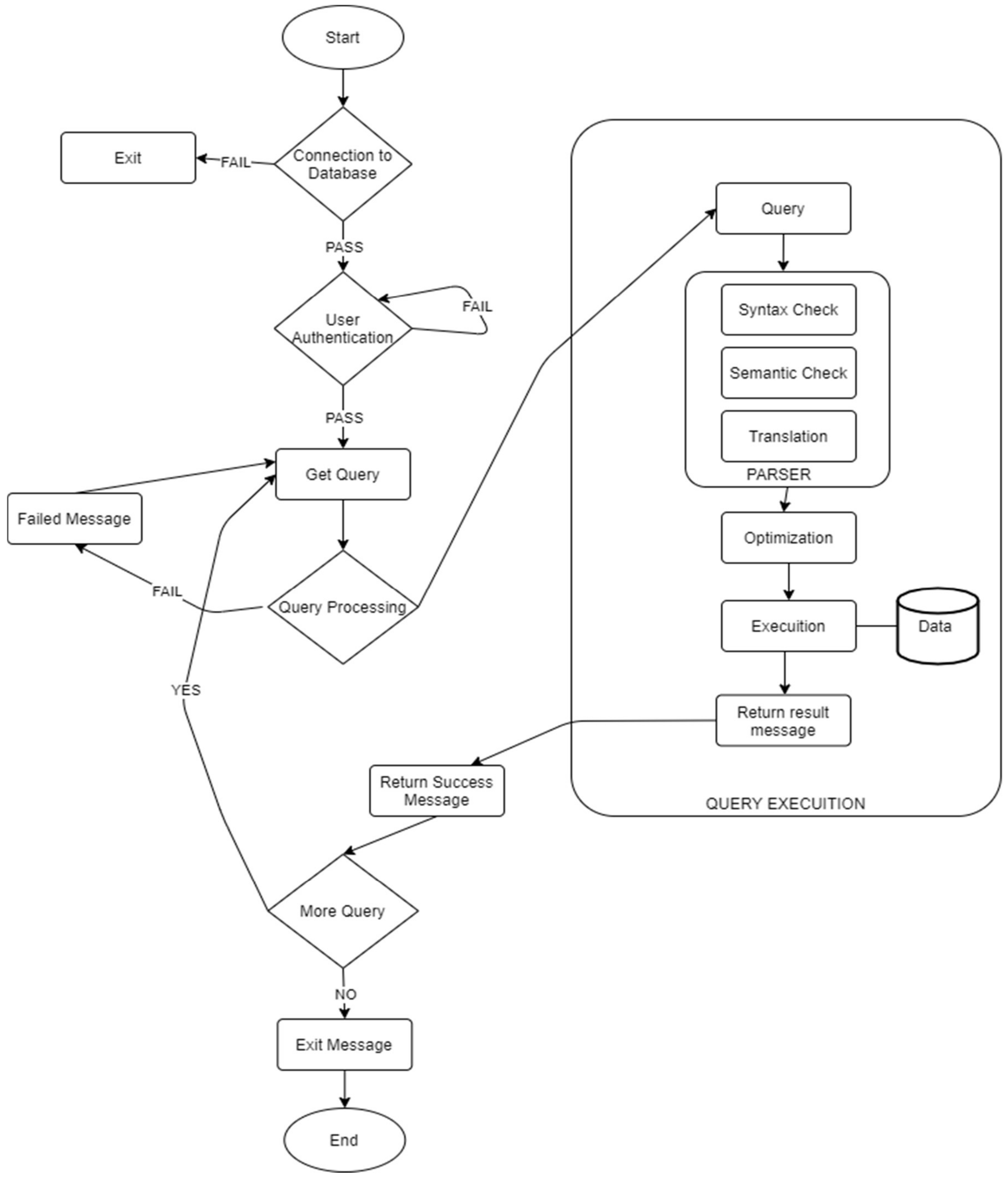


Figure 1: Initial Flowchart for the custom RDBMS

INITIAL PSEUDOCODE:

The flow of the program and a few initial algorithms have been described below in the form of a pseudocode. Figure - 2 depicts connection establishment to the database and then validating the user credentials since this is a multiuser system. Figure - 3 shows reading the query operation from user and performing action as per the query operation. For SQL_QUERY type operation, Query would be validated to check whether the input query is in the scope of the program or not (Figure - 4). Once the scope check is performed, query processor is executed to perform the execution as per the Locking mechanism. This is shown in Figure - 5.

```
// Establish connection with DB
conn = connectionToDB()

if conn failed :
 exit()

user_credentials = input()
isValid = userAuth(user_credentials)
```

Figure 2 – Connection & Authentication

```
if isValid is True :
  queryOperation = input()


if queryOperation is SQL_QUERY :
 query = input()
 queryValidation = validateQuery(query)
if queryValidation is VALID :
 queryType = getQueryType(query)
 queryProcessor(query, queryType)

if queryOperation is GENERATE_SQL_DUMP :
 sqlDump = sqlDumpGeneration()
 display(sqlDump)

if queryOperation is GENEvalidRATE_ERD :
 erd = erdGeneration()
 display(erd)
```

Figure -3 Query Validation and SQL dump, ERD generation

```
validateQuery(query):
if query == ‘SELECT’ OR ‘INSERT’ OR ‘UPDATE’ OR ‘CREATE’ OR ‘ALTER’ OR ‘DELETE’ :
 return VALID
```

Figure -4 Checking the scope of the Query

```
queryProcessor(query, queryType):
 if queryType is 'SELECT':
 rows = runSelectParser(query)
 display(rows)
 if queryType is 'INSERT' OR 'UPDATE' OR 'CREATE' OR 'ALTER' OR 'DELETE' :
         // to implement the 2-phase locking process
         lockResources()
         runParser(query)
         dbCommit()
         unLockResources()
 if queryExecution is success :
         return SUCCESS
```

Figure -5 Two Phase Locking

INITIAL TEST CASES:

Scenario	Input	Output
Connection to db	-	Connection Failed
Invalid authorization	<User Credentials>	Credentials are Invalid

Keyword Spell check	Selact * from abc;
	
	Select * form abc;
		Keyword SELECT is spelled wrong.
		Keyword FROM is spelled wrong.

Table does not exist	Select * from abc;	Table ‘abc’ does not exist
Duplicate table creation	Create table \`abc\` (user_id varchar (50) primary key);	Table ‘abc’ already exist
Duplicate value in Primary key	Insert into abc (user_id) values (‘B00883311’)	This record already exists, duplicate primary key is not allowed
Null value in primary key	Insert into abc (user_id) values (null)	Null values in the primary key not allowed
Data type violation	Insert into abc (user_id) values (00883311)	Int datatype is not allowed in varchar
Invalid format	Select * abc;	It misses the from keyword
Table name as keyword	Create table table (user_id varchar (50) primary key);	Having table name of keyword is not allowed

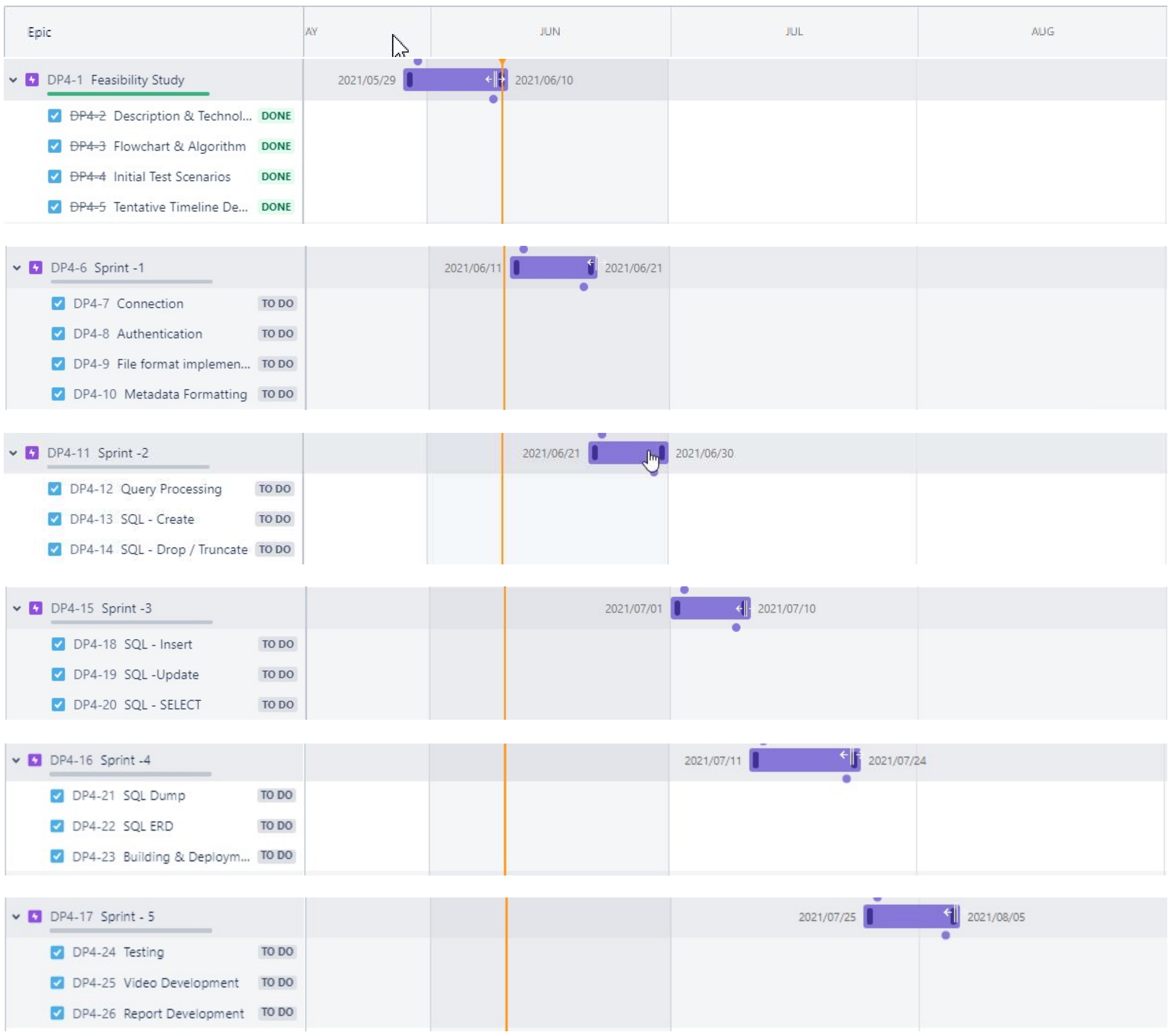
ASSUMPTIONS:

There will be a few assumptions based on which we will plan our implementation.

- Application will accept the input in the form of SQL query only.
- Complex query structures like JOIN will not be supported.
- Only allow WHERE clause is allowed in SELECT and UPDATE query.

TIMELINE:

The Gantt Chart depicting our tentative timeline is displayed below.



Epic | AY | JUN | JUL | AUG

DP4-1 Feasibility Study 2021/05/29 – 2021/06/10
- ~~DP4-2~~ Description & Technol... DONE
- ~~DP4-3~~ Flowchart & Algorithm DONE
- ~~DP4-4~~ Initial Test Scenarios DONE
- ~~DP4-5~~ Tentative Timeline De... DONE

DP4-6 Sprint -1 2021/06/11 – 2021/06/21
- DP4-7 Connection TO DO
- DP4-8 Authentication TO DO
- DP4-9 File format implemen... TO DO
- DP4-10 Metadata Formatting TO DO

DP4-11 Sprint -2 2021/06/21 – 2021/06/30
- DP4-12 Query Processing TO DO
- DP4-13 SQL - Create TO DO
- DP4-14 SQL - Drop / Truncate TO DO

DP4-15 Sprint -3 2021/07/01 – 2021/07/10
- DP4-18 SQL - Insert TO DO
- DP4-19 SQL -Update TO DO
- DP4-20 SQL - SELECT TO DO

DP4-16 Sprint -4 2021/07/11 – 2021/07/24
- DP4-21 SQL Dump TO DO
- DP4-22 SQL ERD TO DO
- DP4-23 Building & Deploym... TO DO

DP4-17 Sprint - 5 2021/07/25 – 2021/08/05
- DP4-24 Testing TO DO
- DP4-25 Video Development TO DO
- DP4-26 Report Development TO DO

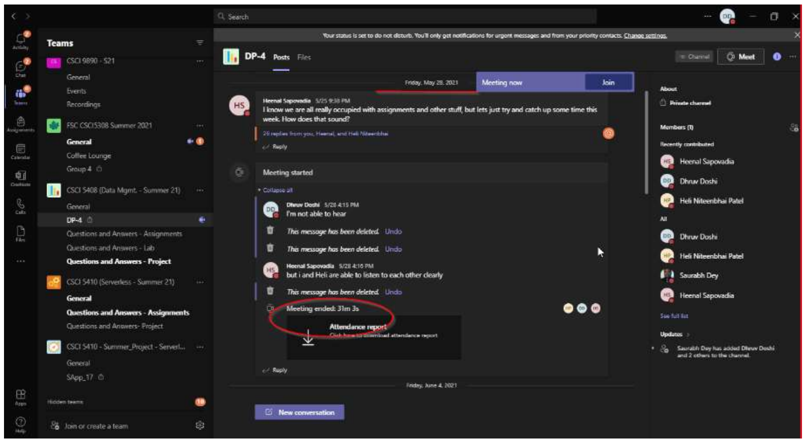
Figure – 6 Timeline

MEETING LOGS:

We have done four meeting in total since the project definition was assigned. During each meeting we discussed various topics about the project.

Meeting _ 1

- Date: 28th May 2021
- Agenda: Icebreaker and initial discussion
- This was the first meeting, so we introduced each other and got to know each other. We also discussed about our known languages, technical abilities, and work experiences. We started with the project requirements in this meeting a bit.


Your status is set to do not disturb. You'll only get notifications for urgent messages and from your priority contacts. Change settings.

Friday, May 28, 2021

Heenal Sapovadia 5/25 9:38 PM
I know we are all really occupied with assignments and other stuff, but lets just try and catch up some time this week. How does that sound?

Meeting started

Dhruv Doshi 5/28 4:15 PM
I'm not able to hear

This message has been deleted. Undo

This message has been deleted. Undo

Heenal Sapovadia 5/28 4:16 PM
but i and Heli are able to listen to each other clearly

This message has been deleted. Undo

Meeting ended: 31m 3s

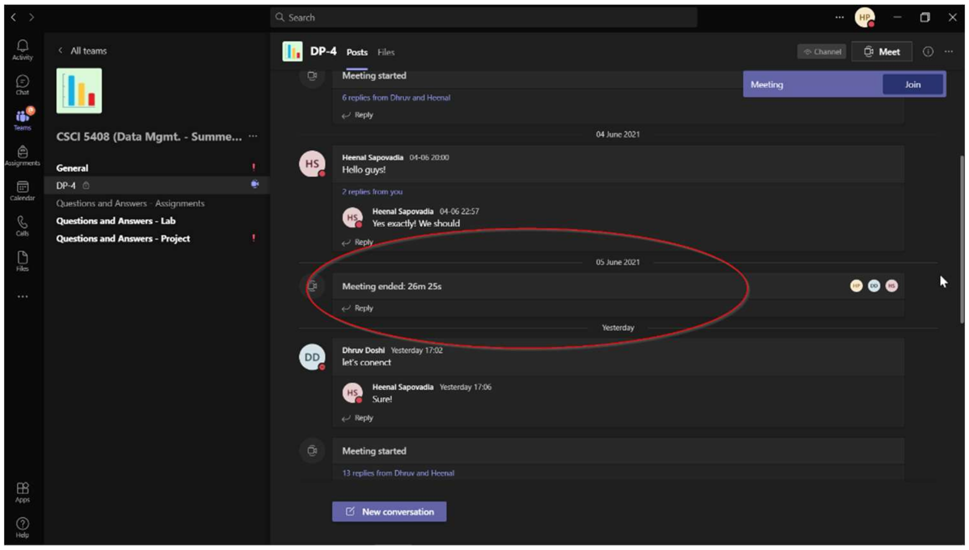
Attendance report

Friday, June 4, 2021

Figure – 7 Meeting-1

Meeting _ 2

- Date: 5th June 2021
- Agenda: Phase 1 discussion
- The 2nd meeting's agenda was to discuss technologies and data structures required for the project. We explored various custom data structure for in memory operations in java. We decided to use HashMap to store the data being read from the persistent storage. After discussion and exploring sites, we narrowed down data structures such as Array List, LinkedList etc. to meet our project requirements.


Search
All teams
CSCI 5408 (Data Mgmt. - Summe...
General
DP-4
Questions and Answers - Assignments
Questions and Answers - Lab
Questions and Answers - Project
DP-4 Posts Files
Channel
Meet
Meeting Join
New conversation

Meeting started
6 replies from Dhruv and Heenal
Reply

04 June 2021

Heenal Sapovadia 04-06 20:00
Hello guys!
2 replies from you
Heenal Sapovadia 04-06 22:57
Yes exactly! We should
Reply

05 June 2021

Meeting ended: 26m 25s
Reply

Yesterday

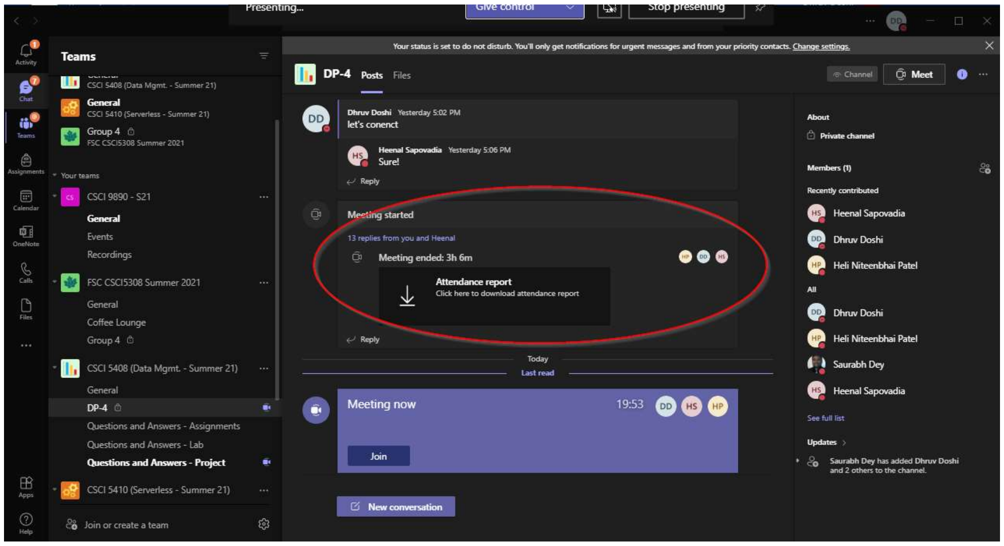
Dhruv Doshi Yesterday 17:02
let's conenct
Heenal Sapovadia Yesterday 17:06
Sure!
Reply

Meeting started
13 replies from Dhruv and Heenal

Figure – 8 Meeting-2

Meeting _ 3

- Date: 9th June 2021
- Agenda: Deciding algorithm
- In the 3rd meeting we discussed about the algorithm that we can use for project development. We should decide the initial flow of the project for the better understanding of the project. So, each of us gave our perspective about project flow and approach by which we can develop the project.



Your status is set to do not disturb. You'll only get notifications for urgent messages and from your priority contacts. Change settings.

Dhruv Doshi Yesterday 5:02 PM
let's conenct
Heenal Sapovadia Yesterday 5:06 PM
Sure!
Reply

Meeting started
13 replies from you and Heenal
Meeting ended: 3h 6m
Attendance report
Click here to download attendance report
Reply

Today
Last read

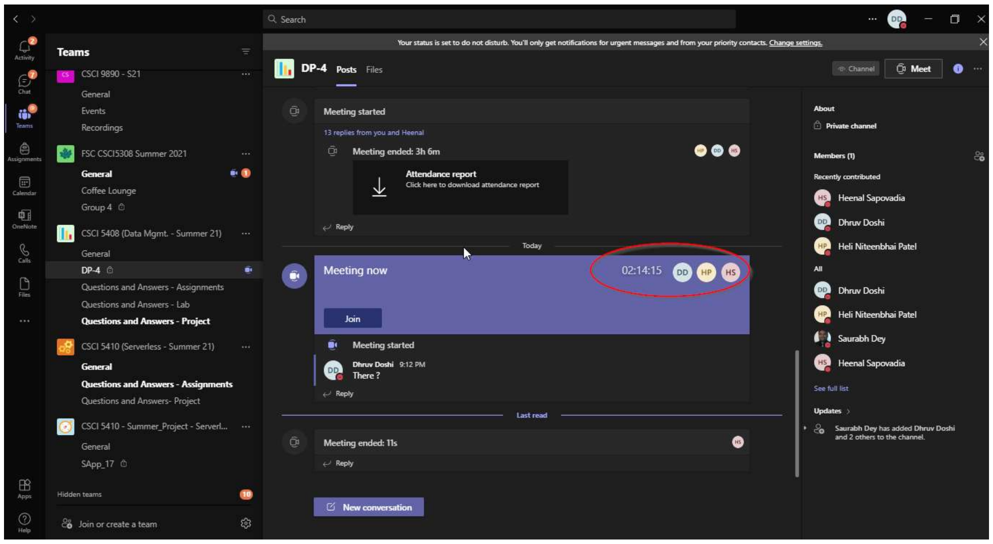
Meeting now 19:53
Join

About
Private channel
Members (1)
Recently contributed
Heenal Sapovadia
Dhruv Doshi
Heli Niteenbhai Patel
All
Dhruv Doshi
Heli Niteenbhai Patel
Saurabh Dey
Heenal Sapovadia
See full list
Updates
Saurabh Dey has added Dhruv Doshi and 2 others to the channel.

Figure – 9 Meeting-3

Meeting _ 4

- Date: 10th June 2021
- Agenda: Feasibility report
- In this last meeting we discussed final feasibility report and completed it.


Teams
CSCI 9890 - S21
General
Events
Recordings
FSC CSCI5308 Summer 2021
General
Coffee Lounge
Group 4
CSCI 5408 (Data Mgmt. - Summer 21)
General
DP-4
Questions and Answers - Assignments
Questions and Answers - Lab
Questions and Answers - Project
CSCI 5410 (Serverless - Summer 21)
General
Questions and Answers - Assignments
Questions and Answers- Project
CSCI 5410 - Summer_Project - Serverl...
General
SApp_17
Hidden teams
Join or create a team
Search
DP-4
Posts
Files
Channel
Meet
Join
Reply
New conversation

Your status is set to do not disturb. You'll only get notifications for urgent messages and from your priority contacts. Change settings.

Meeting started
13 replies from you and Heenal
Meeting ended: 3h 6m
Attendance report
Click here to download attendance report

Today

Meeting now 02:14:15

Meeting started
Dhruv Doshi 9:12 PM
There ?

Last read

Meeting ended: 11s

About
Private channel
Members (1)
Recently contributed
Heenal Sapovadia
Dhruv Doshi
Heli Niteenbhai Patel
All
Dhruv Doshi
Heli Niteenbhai Patel
Saurabh Dey
Heenal Sapovadia
See full list
Updates
Saurabh Dey has added Dhruv Doshi and 2 others to the channel.

Figure – 10 Meeting-4

REFERENCES:

[1]"Flowchart Maker & Online Diagram Software", Draw.io, 2021. [Online]. Available: http://draw.io/. [Accessed: 09- Jun- 2021]

[2]"ArrayList vs LinkedList in Java - GeeksforGeeks", GeeksforGeeks, 2021. [Online]. Available: https://www.geeksforgeeks.org/arraylist-vs-linkedlist-java/. [Accessed: 10- Jun- 2021]

[3]"Lesson: Introduction to Collections (The Java™ Tutorials > Collections)", Docs.oracle.com, 2021. [Online]. Available: https://docs.oracle.com/javase/tutorial/collections/intro/index.html. [Accessed: 10- Jun- 2021]

[4]"Metadata in relational databases (RDBMS) - Database Design & Metadata", Dataedo.com, 2021. [Online]. Available: https://dataedo.com/kb/databases/all/metadata. [Accessed: 10- Jun- 2021]

[5]"ER Diagram: Entity Relationship Diagram Model | DBMS Example", Guru99.com, 2021. [Online]. Available: https://www.guru99.com/er-diagram-tutorial-dbms.html. [Accessed: 10- Jun- 2021]

[6]"Products | Atlassian", Atlassian, 2021. [Online]. Available: https://www.atlassian.com/software. [Accessed: 10- Jun- 2021].