

# C To C: A Web Based Tool For Music Bands

<sup>[1]</sup>Yugandhara A. Nimbalkar, <sup>[2]</sup>Justus Selwyn

<sup>[1][2]</sup>School of Computing Science and Engineering VIT University (Chennai Campus).

<sup>[1]</sup>nimbalkaryugandhara.ajitsinh2015@vit.ac.in, <sup>[2]</sup>justus.s@vit.ac.in.

*Abstract: Musical band tool is designed for music bands for the collection of data from various sources and to organize this data for the bands' management. This tool is a multipurpose organizer that is used for management of bands, the musicians in each of the bands, and planning the concert & practice sessions. It also gives the information about the latest upcoming events. This tool holds the details of all musicians, bands and their concerts. Musicians can also share the related information and audio clips using this tool for further enhancing their knowledge and keeping them up-to-date. This tool tries to cover up different modules related to the musical bands. The tool is dynamic and user-friendly. It is asynchronous in communication, thus making it responsive.*

**Keywords-** Web based tool, AngularJS, NodeJS, MongoDB, ExpressJS, and Client-Server Model.

## I. INTRODUCTION

Today, management and planning is taking importance in common man's life. Everything should be planned and kept organized so that people associated to one's life need not waste their time and money. Many ideas are coming up for Management Systems in different fields. Basically, sky is the limit for developing Management Systems using Tools. These kinds of systems have proved to be a lot of help for big organization where many different people are involved in achieving a common goal. The data is collected from various sources and saved on a shared server database. The same data can then be retrieved from this shared database. This kind of system is also called client-server model. In this model, client sends the requests to the server about obtaining some data or storing some data. The simple client-server architecture is given in figure 1. Accordingly, the request is processed with the appropriate response, which is sent back to the client. The server is kept in secured remote area and users requesting the server are the clients.

This server can be used to for data insertion, retrieval, updating and deletion. Along with this some server are doing the analysis of the data which is stored in the database as well. Many mining and learning techniques are available. These techniques either work for Business Intelligence or Prediction purposes.

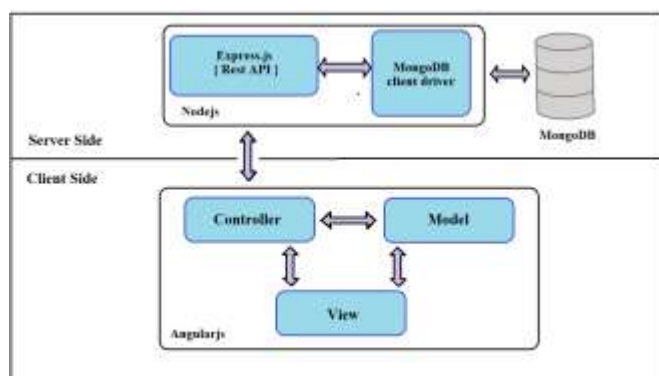


Figure 1. Client Server Model

## II. LITERATURE SURVEY

F. J. Muñoz-Rodríguez and G. Jiménez-Castillo have designed, developed and implemented an app which manages to control an electronic lab bench [2]. The implementation has been done using Java. This can prove a good project in E-learning. Amjad Farooq and Imran Mir have designed and developed a web tool for University [1]. The proposed tool covers the important aspects of students and teacher and has got some new features which are different than the traditional web applications.

Yunhua Gu, Shu Shen, Jin Wang and Jeong-Uk Kim give the application of MongoDB in Information Collection System [3]. They have proved that the relational database system having multiple tables and each of them having foreign key have low query performance. Also, they have high cost of vertical scalability. There is inconsistency in Object-Relation mapping. Abdur Rahman.A and Chitra Devi S. have developed a web based content management system which carries the advantages of accessibility, consistency and security [4]. This software is developed using AngularJS and Joomla. Ravi Sharma and Shipra Ravi Kumar have developed the strategies for Web Application development process [7]. They have discussed these methodologies based on agile development approach. One of the approaches for web application development discussed was UML based. It is easy and effective approach to manage and control development process by the development of UML diagrams.

### III. PROPOSED WORK

In the proposed work, the web based tool is created for the Music Bands across the country. The users who will be using this tool are administrator, band leader, musicians, Event Organizers and band manager. Each of them have been assigned with the different access rights. Accordingly, their views will change. At first, the admin is created and given all the access rights. He can view the details of all the users. He can also see the events created by users and the transactions taking place between two users. Event organizers and Band leader have the rights to view their own as well as the users' profile under him as shown in figure 3 and 4 respectively. While the band manager and musicians can view only their profile. If the band manager is the band leader himself then he still can view the other musician profile. The following sub-sections present detailed description of access rights of the actors.

#### *Administrator*

The administrator is the overall controller of the tool's operations. The access rights provided for the administrator is that he can view all users profile and details of the users' transactions and events. He can approve or disapprove the accounts created by any other users.

He can broadcast some important messages to the different users group or all of them. If an account is inactive since long time, then he has the right to delete that account permanently from the database. The class diagram for administrator is given in figure 2.

#### *Event Organisers*

The event organiser is the one who organises events for music bands. He can invite the music bands as per the style of the music bands and the occasion. The creation of event has the fields like name of event, venue, start time, end time and so on. The methods of this functionality include create and broadcast the event. After the event is finished, he needs to create the report of the event and gives reviews about the participated bands. The fields included in this functionality include the number of bands participated, the number of bands performed, the names of bands participated and the names of bands performed. This will help other event organisers to select and invite those Bands. The class diagram for Event-Organizer is given in figure 3.

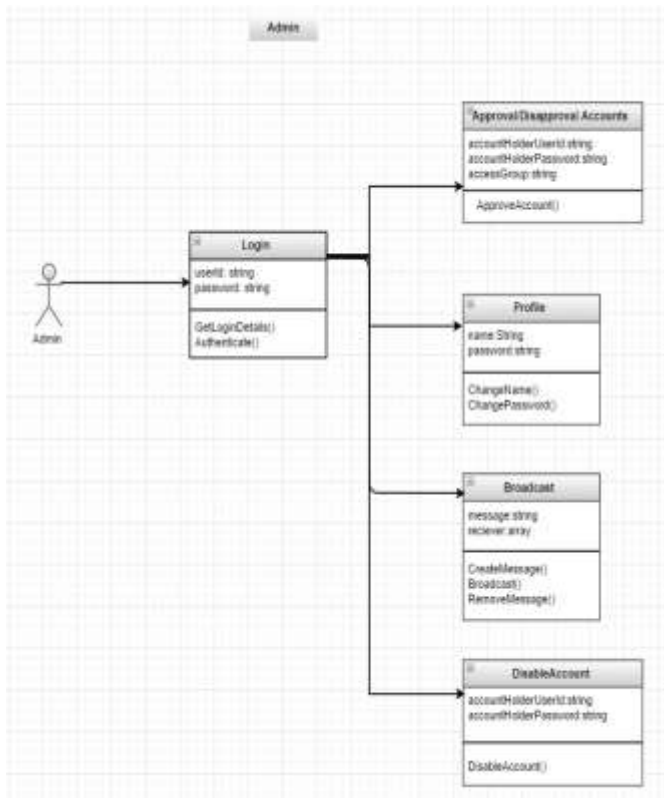


Figure 2. Administrator's class diagram

### ***Band Leader***

He can choose one of the musicians as the band manager or he can choose himself or somebody else who is not a musician.

He also has the right to create events and invite other bands for the event. He has the rights to share the necessary artefacts (in the pdf format) with the band members. He can chat with his band members. He also has the rights to schedule the practice session for the upcoming events. He can view the detailed plan of the event created by the band manager for the band. The class diagram for band-leader is given in figure 4.

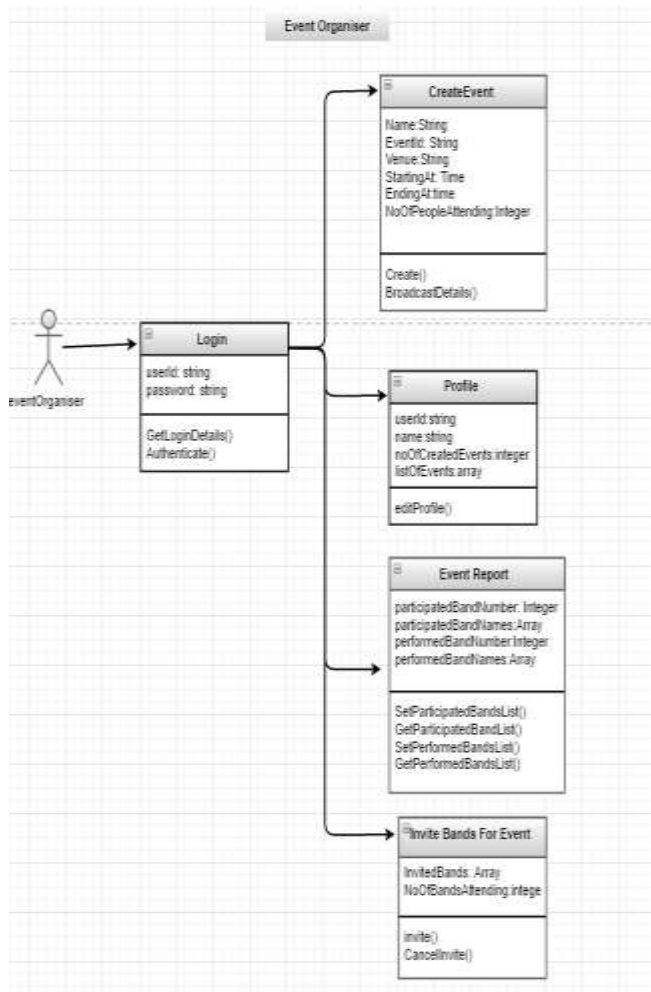


Figure 3. Event Organiser's class diagram

### ***Band Manager***

Band manager is the one who works on the technical aspect of the band and helps the band leader in his work. It is band leader's wish to select a different person as band manager or to work on technical aspects all by himself. Depending on this, the band manager's access rights are separated from the band leader's ones. The class diagram for band-manager is given in figure 5.

Band manager has the rights to upload the videos and images on behalf of the band regarding the event which has occurred. The fields included in this functionality are image or video file. The methods included are add video/image and remove video/image.

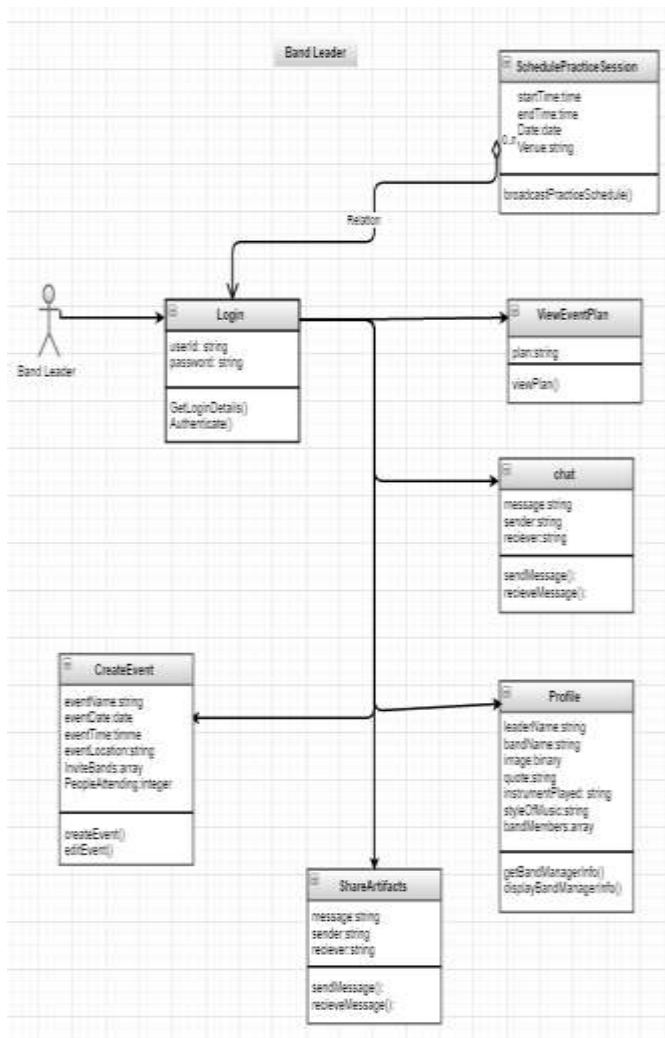


Figure 4. Band Leader's class diagram

***Musician/ Band member***

Band members and musicians together with their band member perform at the events. They play different instruments and form a team. They can view the detailed plan of the event sent by their band manager. The fields included in this functionality are plan and method is view the plan. They can chat with the other group members and share artefacts in the form of pdf. They can share audio and video necessary for their rehearsals. The fields for this functionality sender, receiver, and the audio or video file. The class diagram for band-member is given in figure 6.

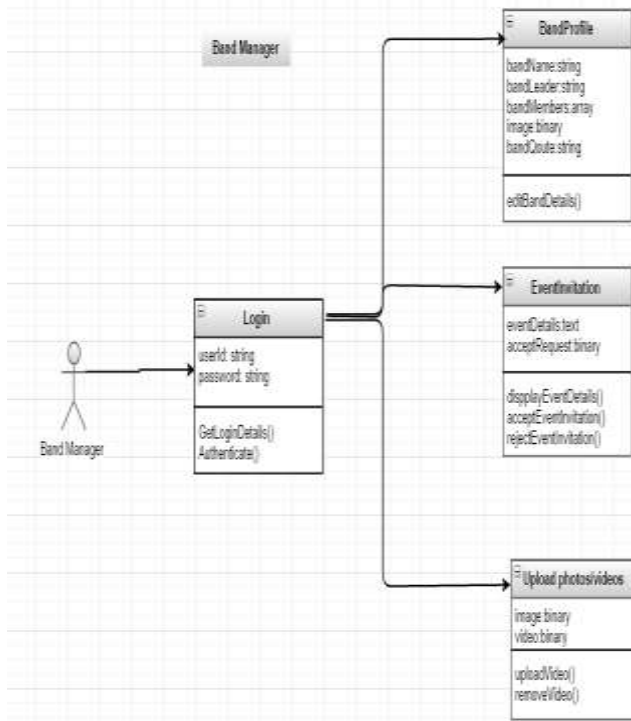


Figure 5. Band manager's class diagram

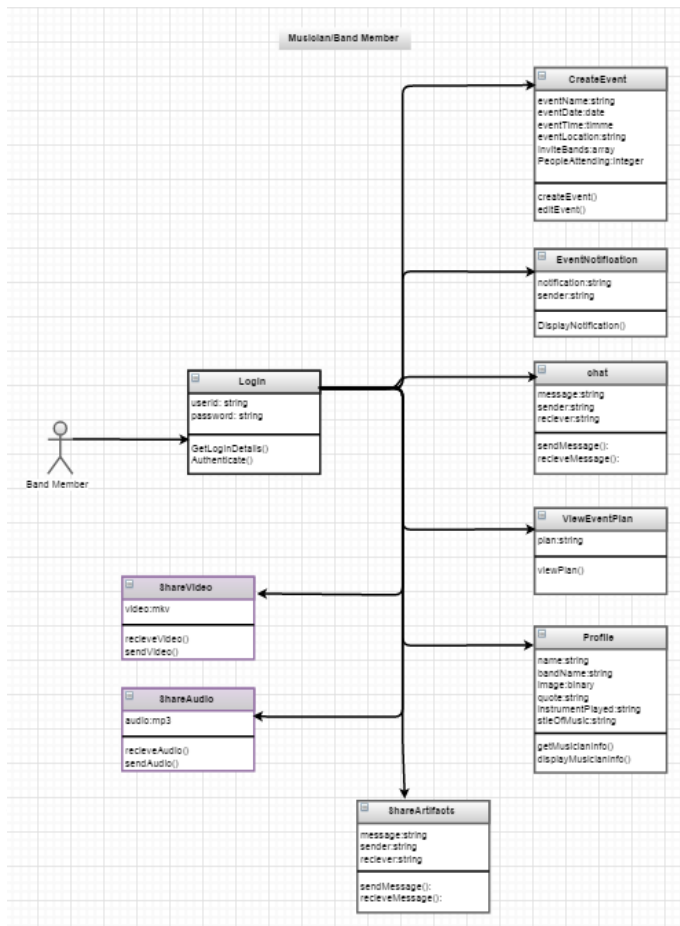


Figure 6. Band member's class diagram

**Software package used:**

For this web tool, MEAN Stack is used. MEAN stands for M-MongoDB, E-Express, A- AngularJS and N-Node.js [6]

1. MongoDB- It is document oriented database system. It stores the data in the form of collections of documents [5]. The documents contain fields which are the elements of data.
2. Express- Express also called as ExpressJS is used at the backend and is the framework of Node.js. It is written in Javascript.
3. AngularJS- AngularJS along with HTML is used at the frontend. This is an open source software and is also written in Javascript. AngularJS can create light-weight, user-friendly and dynamic applications.
4. Node.js- It is an open source software and is used at the backend along with MongoDB. This can also be used to develop APIs.

At first, the server is created on the backend using NodeJs. In that web server is created using ExpressJs. The other necessary modules for authentication, mongoose, and development of log file, images and the other modules developed for routing of the routes are created. The mongoose module is necessary to connect NodeJs with MongoDB. The Schema is created using mongoose and this Schema is converted to models to put them into database. Each document in the database can be managed using CRUD (create, read, update and delete) commands.

The database of MongoDB contains collections of each of the users like band manager, band leader, band member, event organiser and admin. Also it has the collections of instruments played, genre, events created and invitations.

The client-side developed in AngularJs has model-view-controller. In AngularJs, the app is created that controls all the functionalities in the front-end. The \$scope variable is used to create the model on the client-side. The view is created using HTML and controller is created using AngularJs. Each view is connected to the controller. The controller is the processing unit of the MVC model.

On the client-side, the template is set by the viewer container which is the basic view of the tool. The others views are injected in this view because this is a single page application.

#### IV. RESULTS

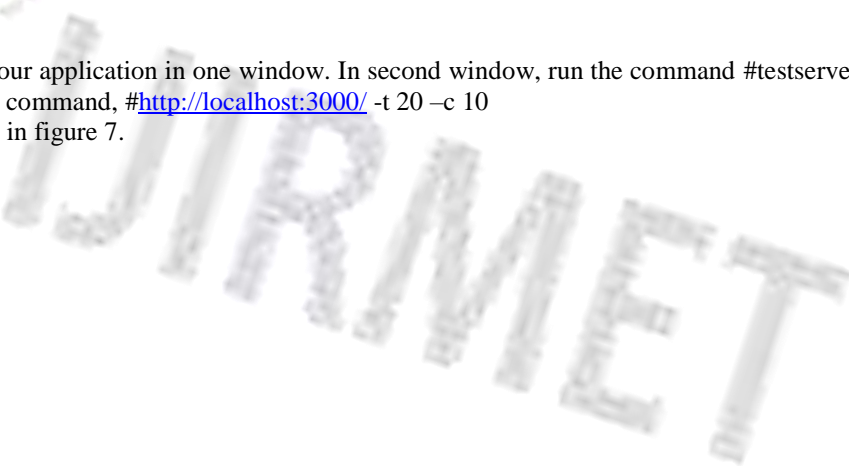
C to C: A Web based Tool for Music Bands is analysed for number of requests sent in per second. The evaluation may also be carried over a period of time with number of different requests.

To analyse this, a module is installed in NodeJS called loadtest. This is used to run the load test on particular HTTP requests or URLs. For this, loadtest is needed to be installed. Following command is used for installing the loadtest:

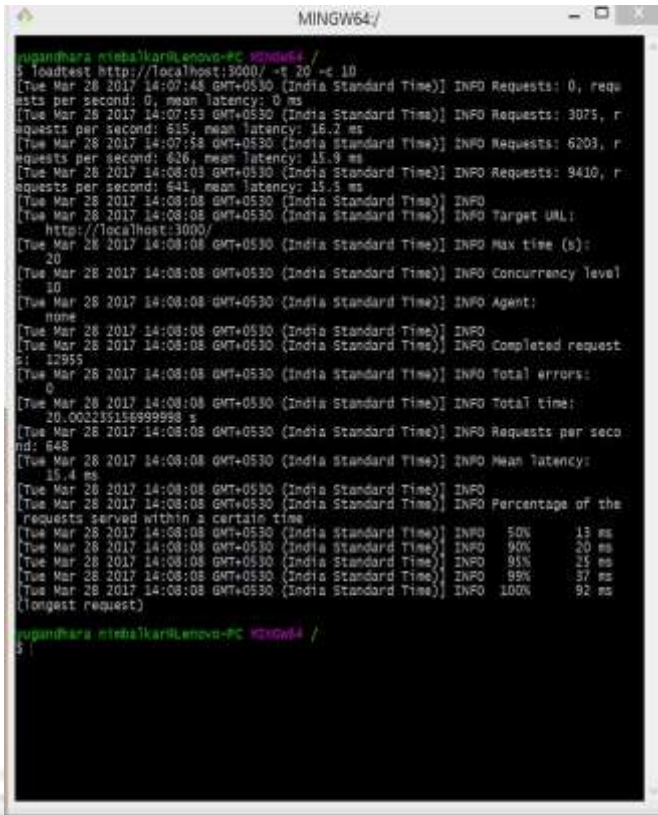
```
# npm install -g loadtest --save
```

Open 3 windows of command prompt. Run our application in one window. In second window, run the command #testserver-loadtest loadtest and in third window run the command, #<http://localhost:3000/> -t 20 -c 10

This will generate the analysis as shown in figure 7.







```

MINGW64/
C:\Users\rugandhara_niteba\karilamovo-PC> loadtest http://localhost:3000/ -r 20 -c 10
[Tue Mar 28 2017 14:07:48 GMT+0530 (India Standard Time)] INFO Requests: 0, requests per second: 0, mean latency: 0 ms
[Tue Mar 28 2017 14:07:53 GMT+0530 (India Standard Time)] INFO Requests: 3075, requests per second: 615, mean latency: 16.2 ms
[Tue Mar 28 2017 14:07:58 GMT+0530 (India Standard Time)] INFO Requests: 6203, requests per second: 626, mean latency: 15.9 ms
[Tue Mar 28 2017 14:08:03 GMT+0530 (India Standard Time)] INFO Requests: 9410, requests per second: 641, mean latency: 15.5 ms
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO Target URL: http://localhost:3000/
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO Max time (s): 20
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO Concurrency level: 10
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO Agent: none
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO Completed requests: 12955
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO Total errors: 0
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO Total time: 20.002735156999898 s
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO Requests per second: 648
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO Mean latency: 15.4 ms
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO Percentage of the requests served within a certain time
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO 50% 13 ms
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO 90% 20 ms
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO 95% 25 ms
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO 99% 37 ms
[Tue Mar 28 2017 14:08:08 GMT+0530 (India Standard Time)] INFO 100% 92 ms (longest request)
C:\Users\rugandhara_niteba\karilamovo-PC>
  
```

Figure 7. Analysis of load screen shot

Table 1. Requests and Mean latency

Requests	Requests per second	Mean Latency
3075	615	16.2 ms
6203	626	15.9 ms
9410	641	15.5 ms

Table 1 gives the analysis of total requests sent, requests sent per second and mean latency between 2 requests.

Table 2. The result obtained in load test

Target URL	http://localhost:3000/
Max time(s)	20
Concurrency level	10
Completed requests	12955
Total errors	0

Total time	20.002235156999998 s
Mean latency	15.4 ms

Table 2 gives the analysis about some other features like Target URL, Maximum time required, concurrency level, the number of requests completed, total number of errors, total time required and mean latency which are obtained from the server.

Table 3. Percentage of requests and time period

Percentage of Request	Time Period
50%	13 ms
90%	20 ms
95%	25 ms
99%	37 ms
100%	92 ms

Table 3 shows the percentage of requests and the time period required for their execution.

## V. CONCLUSION

A Web tool is designed and implemented which can help the Music Bands and various Event Organizers to connect with each other using a client-server model. The application is a single-page application, thus making it faster for its response. The software used in Web Tool is MEAN stack which helps create the light-weight application. Nodejs is used for developing the APIs on the server-side. While AngularJs is used on the client-side. The database used is MongoDB which is the document-oriented database. The language used CSS or Cascading stylesheets and the framework of Bootstrap makes the application user-friendly and responsive.

Various access groups are created in the application. They include Admin, Band Members, Band Managers and Event Organisers. Each of the user has different activities and privileges according to these access groups. Each user has a profile. This user has been given privilege to manage his profile.

If the user belongs to Event-Organiser access group, he has the privilege to create new events and send an invitation to bands to perform in his event. The band manager has got the privilege to accept the invitation and give the details of that event to other band members.

The band members can send artefacts and audio or video files to the other band members for preparation of the upcoming events.

The Admin can monitor all the activities which are going on using this application. The Admin has the privilege to remove the user from the database for being inactive for a long time or for misbehavior.

## VI. REFERENCES

- [1] Amjad Farooq and Imran Mir, "Design and Development of a University Web Tools", International journal of multidisciplinary sciences and engineering, vol. 1, no. 2, November 2010.
- [2] F. J. Muñoz-Rodríguez and G. Jiménez-Castillo, "Web app for a Remote Electronics Instrumentation Lab".
- [3] Yunhua Gu, Shu Shen, Jin Wang and Jeong-Uk Kim, "Application of NoSQL Database MongoDB", 2015 International Conference on Consumer Electronics-Taiwan (ICCE-TW)

- [4] Abdur Rahman.A and Chitra Devi S., “A Framework for Ultra-Responsive Light Weight
- [5] Web Application Using Angularjs”, Online International Conference on Green Engineering and Technologies (IC-GET 2015), 2015
- [6] MongoDB database Documentation docs.mongodb.com
- [7] MEAN.io. (2015) MEAN — Full-Stack JavaScript Using MongoDB, Express, AngularJS, and Node.js. [Online]. Available: <http://mean.io/>
- [8] Ravi Sharma and Shipra Ravi Kumar, “Strategies for Web Application Development Methodologies”, International Conference on Computing, Communication and Automation (ICCCA2016), 2016

