**Meeting Zone**

**Team**
- Mustafa Tayyip BAYRAM - 170706007
- Furkan ÖCALAN - 170706003
- Damlanur TOPCU - 170706012
- Ecem YENİTÜRK - 170706024
- Kutay TORLAK - 180706026
- Mert YILMAZ - 180706006

**Week 01: Review**
- Determine the roles of each group member
- Research Open-Source projects and determine whether to use an open-source project or create a project from scratch.
- Select an open-source scrum tool.
- Test selected open-source project and conclude.
- Create a GitHub repository for the project
- Listing the useful-handy resources
- Consider risks:
  - Project Size
  - Implementation Risks
  - Operational Impacts
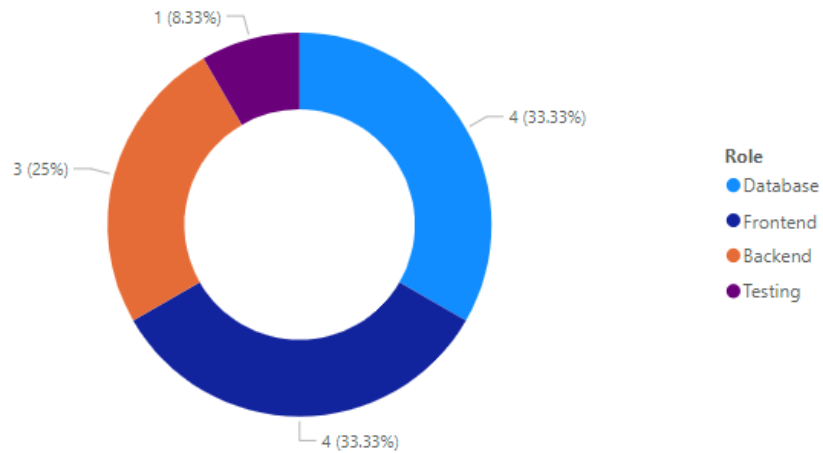
**Week 01 Tasks**

| Team Member | Mustafa | Furkan | Damla | Ecem | Kutay | Mert |
|---|---|---|---|---|---|---|
| Tasks | Creation of GitHub repository ,selecting an open-source Scrum tool | Research for open-source projects and Scrum tool | Testing selected open-source project. | Research and list suitable guides and resources | Indicate the risks | Create charts for report |

**Role Distribution**

| Team Member | Mustafa | Furkan | Damla | Ecem | Kutay | Mert |
|---|---|---|---|---|---|---|
| Role | Backend - Testing | Backend - Database | Frontend - Backend | Frontend - Database | Frontend - Database | Frontend - Database |

Distribution by Role



## Next Week Tasks

| Team Member | Mustafa | Furkan | Damla | Ecem | Kutay | Mert |
|---|---|---|---|---|---|---|
| Tasks | Project Estimation - Scheduling | Risk Plan | Determine the needs of the Stakeholders | Statement of Work | Vision Statement | Scope of phased release |

## Open-Source Projects

| Project | Advantages | Disadvantages |
|---|---|---|
| MeetingPoint | • Does not have many dependencies to install<br>• Easy to use | • Host of the meeting cannot restrict members<br>• Custom accounts cannot be created. |
| Jitsi-Meet | • It nearly has all the features of Zoom<br>• Android/IOS compatible<br>• Build on top of WebRTC | • Hard to install<br>• This project is too complex for developing and moving forward. |
| BigBlueButton | • Doesn`t need special clients to install. It runs inside the web browser | • Too complex like Jitsi-Meet<br>• Has many dependencies |

| Tool | Advantages | Disadvantages |
|---|---|---|
| OpenVidu | • Build on top of WebRTC<br>• Has most of the features of Zoom | • Hard to Install<br>• Too Complex |
| Video-Meeting-App<br>(We choose this) | • Easy to install | • Has minor bugs |

## Open-Source Scrum Tools

| Tool | Advantages | Disadvantages |
|---|---|---|
| YouTrack | • Enables connection with related IDE.<br>• Free for teams of up to 10 | • Too complex for starting. |
| JIRA | • Love how versatile it is, being able to make plan. have tasks depend on each other, assign tickets/tasks, have mini-goals/mini-projects(epics) | • Might be too powerful - it can become extremely complex, complicated, and difficult to understand<br>• Not free |
| **Agilefant** | • Powerful<br>• Free | • Old and outdated user interface for easy to use |
| Trello<br>(We choose this) | • Free<br>• Hosted in the cloud, not standalone<br>• Easy to use | • Storage limit<br>• Problem with commenting |

## Listing the useful-handy resources

Eligible programming languages for this project have been listed into the table.

| Programming Languages – Frameworks |
|---|
| HTML, CSS, Bootstrap |
| JavaScript – NodeJS – ReactJS - EJS |
| SQL |

The group gathered information after looking at such websites. The group explored various ideas that could serve as a guide.
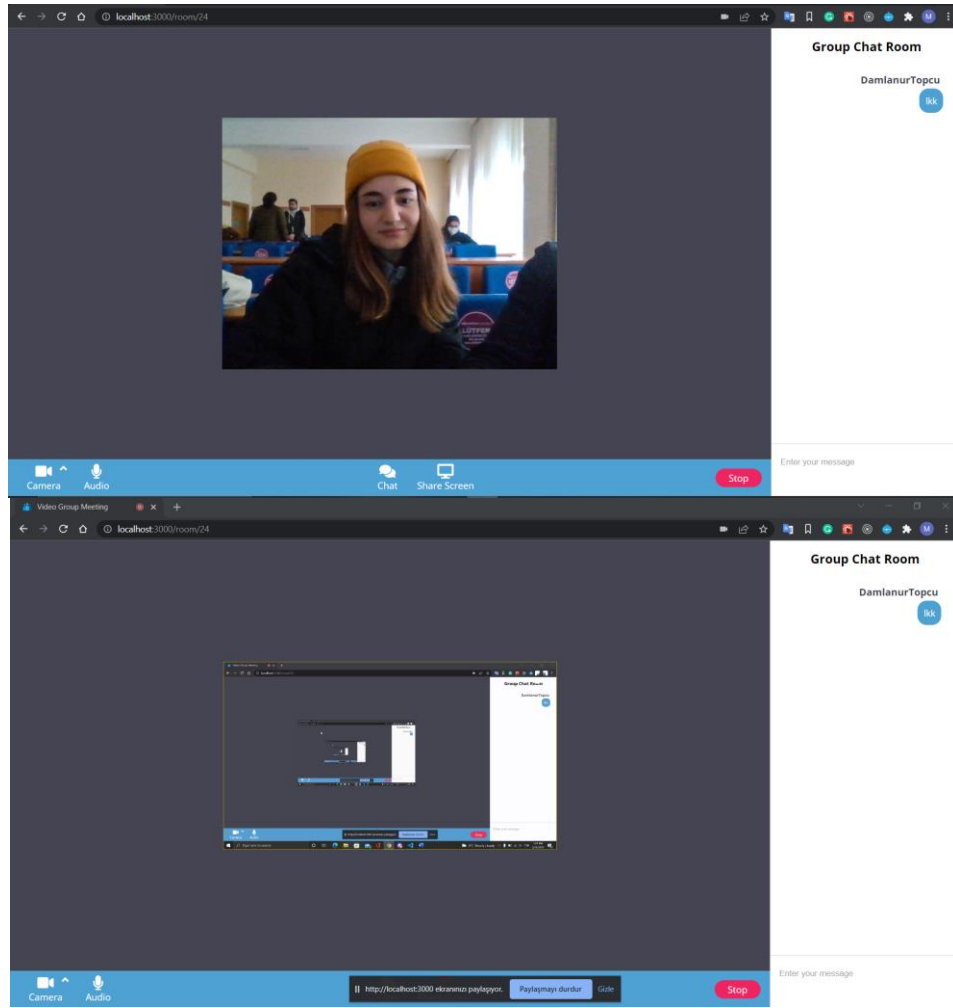
- https://webrtc.ventures/2018/02/learn-how-to-build-a-video-conference-app-and-not-die-trying/
- https://web-and-mobile-development.medium.com/how-to-develop-a-video-conferencing-app-like-zoom-d6ed35dc37e7
- https://www.matellio.com/blog/how-to-build-a-video-chat-app-like-zoom/
- https://codeit.us/blog/how-to-make-a-video-calling-app
- https://www.cleveroad.com/blog/how-to-make-a-video-chat-app

The team then decided which sites were more suitable for the project or more effective for deployment.

## Testing of the Open-Source Project

| Tested Project | Installation Process? | Eligible for our project? | Conclusion |
|---|---|---|---|
| Video-Meeting-App | It was easy to install did not have many dependencies to run | This project seems reasonable to use as a backbone for our project because it has the main features which we need, and it is open to develop further. | As a conclusion, this open-source project meets the requirements to become the backbone for our project. |

This week we found some open-source projects from GitHub but only chose one of them which is called: Video Group Meeting. It is written in JavaScript language. For installation, we downloaded Node.js. Then, inside of it there are 2 files. One of them is client and other one is server. I installed npm in each of them and started it. It worked fine and these are the screenshots from the web app:

**GitHub Repository**

As open source has many benefits, it still depends on which perspective it's looked. In our project, we concluded it to be an open-source project.

- Open-source software's provide an extra security layer because the project can be contributed by people who use it.
- Code can be customized according to users needing's.
- It is more interoperable than the closed source which means it's more compatible with users.

Project codes will be available to public in the following repository:  https://github.com/mutabay/Meeting-Zone

# Software Development Risks

## Bad Timing

- Wrong time estimation,
- Not properly reviewing the resources,
- Unexpected project scope extensions,
- The project manager does not properly track employees, skills, task status.

## Incorrect Budget Estimation

- The budget was initially calculated incorrectly.
- No funds have been reserved.
- Unplanned project expansion.

## Poor Code Quality and Technical Risks

- Lack of knowledge of team members.
- Changes in software requirements.
- Communication weakness in the team.
- Very complex multi-piece project.
- The product is complicated for implementation and the integration of project modules is difficult.

## Poor Management

- Lack of leadership and analytical skills,
- Poor risk management in software development,
- Poor communication and interaction within the team.

## Copyright

- Lack of leadership and analytical skills,
- Poor risk management in software development,
- Poor communication and interaction within the team.

### Ratio by Risks



0.063 (6.29%)
0.313 (31.24%)
0.188 (18.76%)

**Risks**
- Technical Risks
- Bad Tİming
- Budget Estimation
- Poor Management
- Copyright

0.188 (18.76%)
0.25 (24.95%)