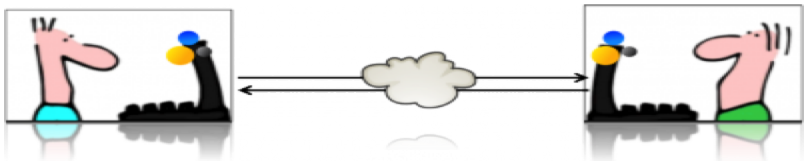
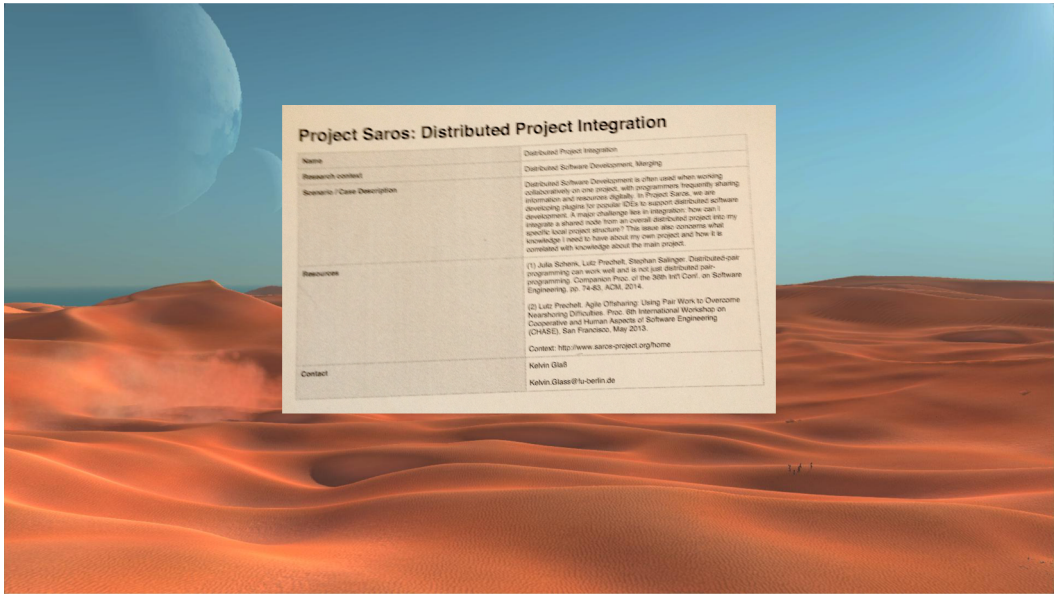
# Saros:
# Distributed Project Integration

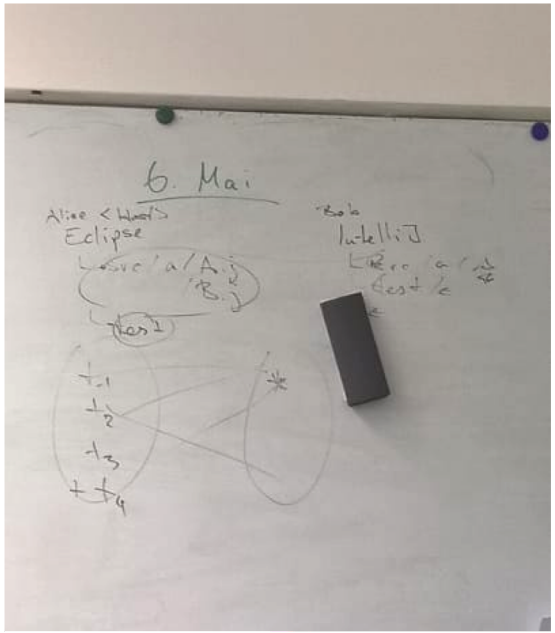
Beniamin Sypianski, Cedric Laier, Mahmoud Kozae



# Knowledge at the beginning



# General Idea

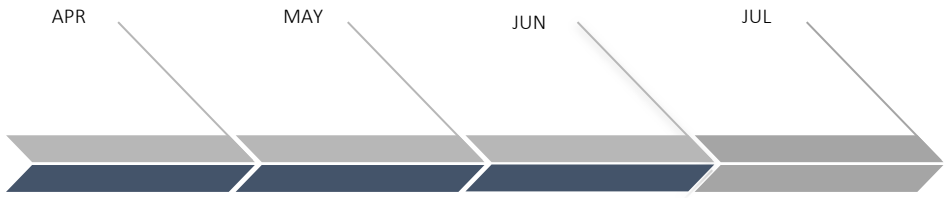



Focus on the project's structure

Important questions:

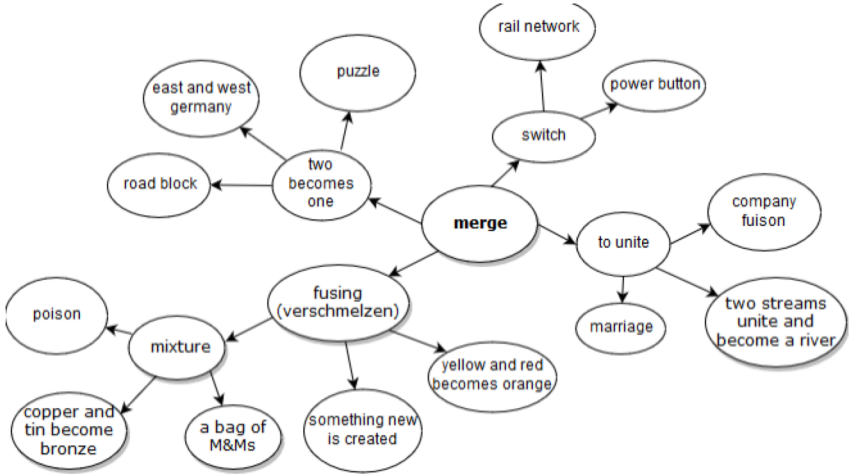
- How to integrate files between participants of a Saros session?
- How to visualize the whole process to the user?

# Project Timeline



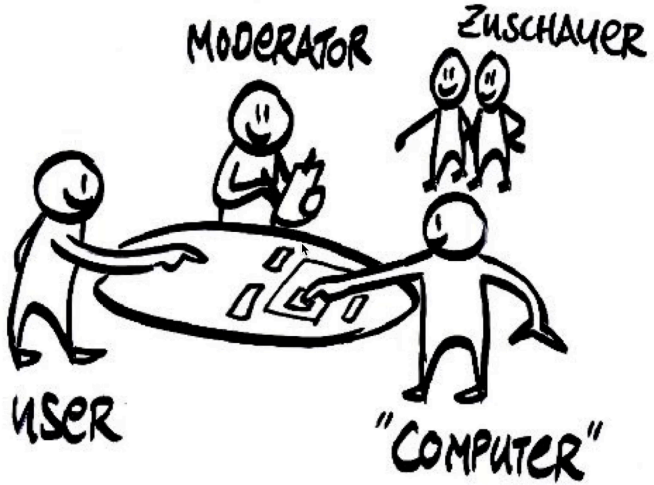

| Initial Research | Brainstorming | Making real | Finishing |
|---|---|---|---|
| Interviewing contributor from the Saros group<br>Secondary research | Taskflow, storyboard<br>Metaphors, analogies<br>Paper prototype | Technology choice<br>HiFi prototyping<br>Formative evaluation | Summative evaluation |

# Creative Part




# Paper Prototype & First Iterations

## Paper Prototypes



# Biggest Challenges

- Find a way to visually compare projects
- Identify possible file & folder states
- Find the the “right” icons and data visualisation to transfer this knowledge to the user
- Come up with a task flow to resolve conflicts

| Possible Folder States | File States in Folder | File Color |
|---|---|---|
| Contains older files | Older File | Grey |
| Contains right change files | Right change file | Purpe |
| Contains left change files | Left change file | Cyan |
| Contains mergeable files | Mergeable file | Orange |
| Contains conflict files | Conflict file | Red |
| Contains right change and older files | Right change & Old file | Purpe |
| Contains left change and older files | Left change & Old file | Cyan / Grey |
| Contains left change and right change files | Left change & Right change file | Cyan / Grey |
| Contains mergeable and older files | Mergeable file & Older file | Orange / Grey |
| Contains mergeable and right change files | Mergeable file & Right change file | Orange / Purpe |
| Contains mergeable and left change files | Mergeable file & Left change file | Orange / Cyan |
| Contains conflict and older files | Conflict file & Older file | Red / Grey |
| Contains conflict and right change files | Conflict file & Right change file | Red / Purpe |
| Contains conflict and left change files | Conflict file & Left change file | Red / Cyan |
| Contains conflict and mergeable files | Conflict file & Mergable File | Red / Orange |

| Merge Actions | Icon | Merge Actions | Icon |
|---|---|---|---|
| *Regardles of merge target:* | | *When mergin to Target:* | |
| No action required | ✓ | Copy to right | → |
| Merge automatic | ↻ | Delete right | x |
| manual merge required | !! | | |
| *When mergin to Source:* | | *When mergin to New:* | |
| Copy to left | ← | Take Source | ↘ |
| Delete left | x | Take Target | ✓ |

# Hi-Fi Prototype

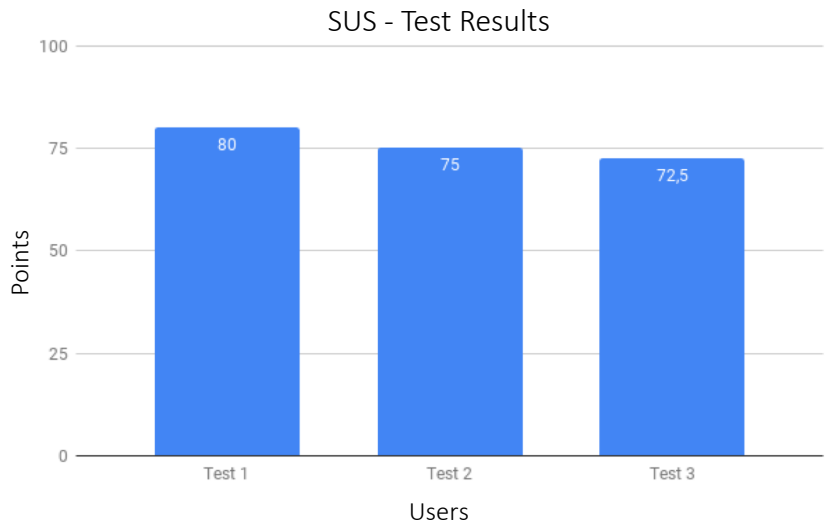
LIVE DEMO

also hosted on a S3 Bucket here

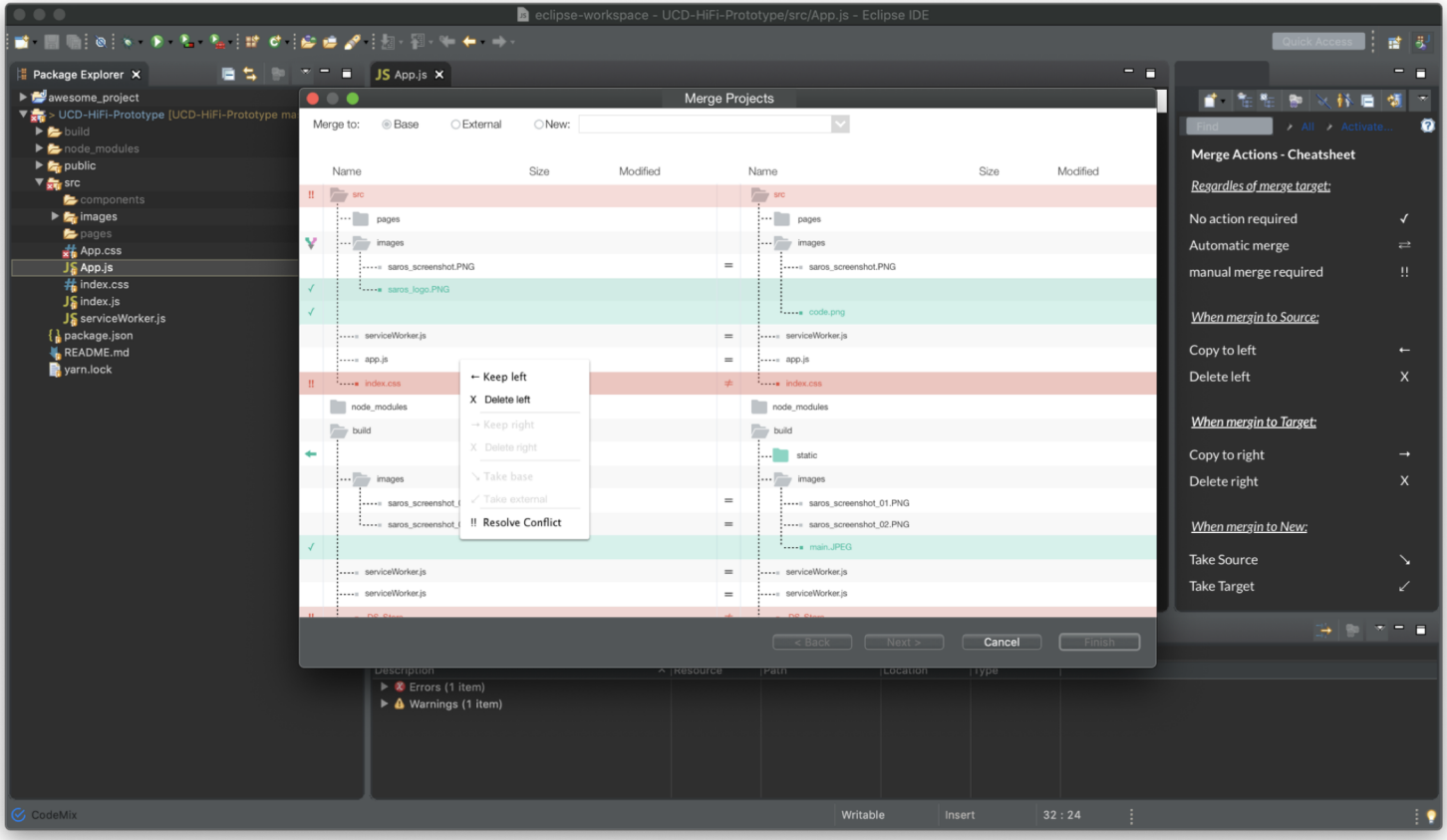
# Testing the Hi-Fi Prototype

## System Usability Scale (SUS)

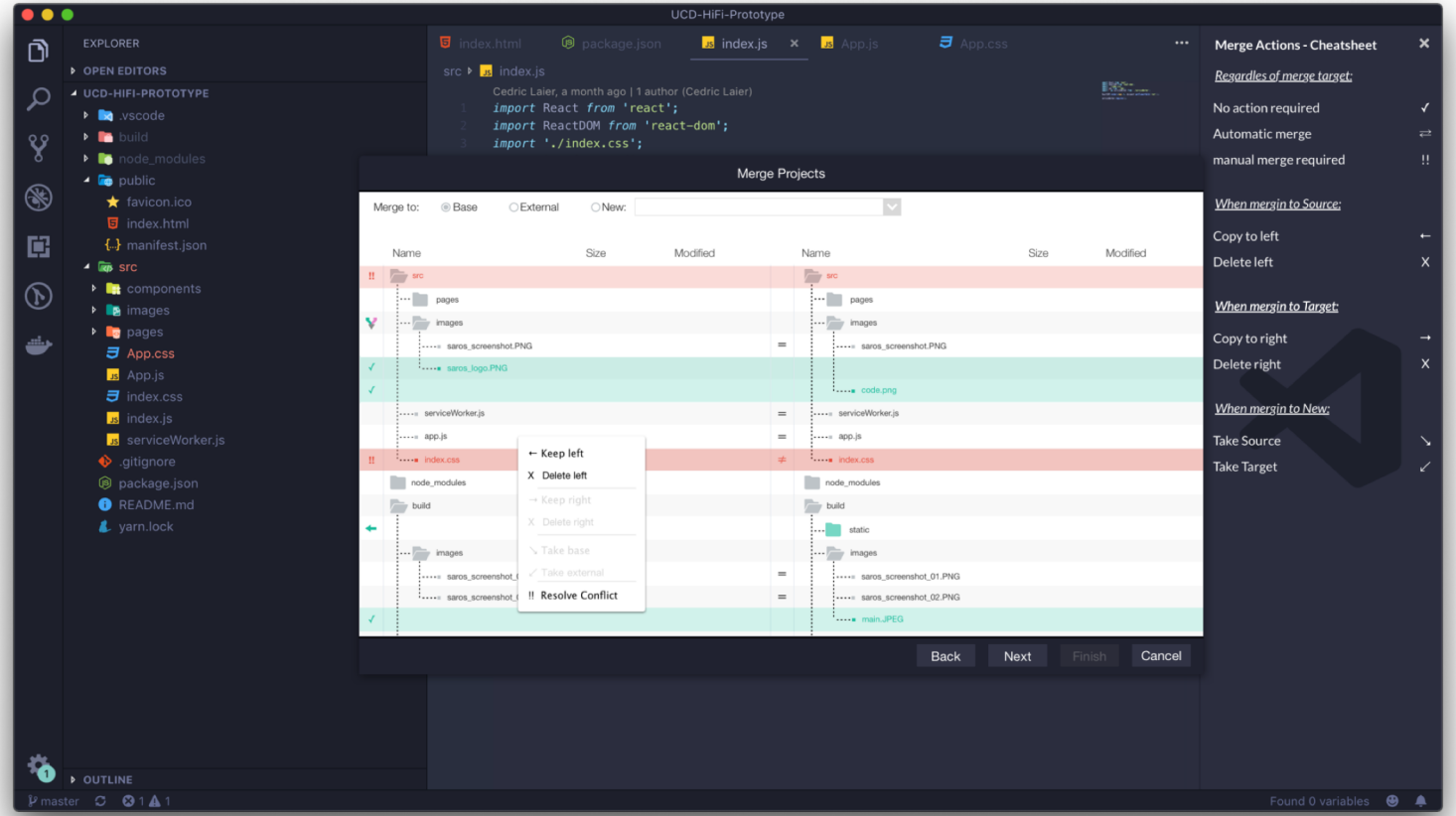
- We wanted to get an high level overview of the view of usability for the user
- Should allow us to quickly compare the effect of adjustments
- Tested professional Software-Engineers
- Overall average Score of 75,6 Points (> 71,4 = Good ; > 85,5 = Excellent)



# Outlook (Eclipse)



# Outlook (VS Code)



# Outlook

What's missing:

- Add more Scenarios: Merge base into external project & Merge base project + external project into new project
- Extend Prototype and have more Test-Iterations
- Finetune Icon mapping
- Add color-codes for folder/file states
- Extend the prototype by a "Resolve Conflicts" View

# Thank you!

All artefacts can be found here:

Images, Sheets, Assignments, Tests etc. : https://drive.google.com/drive/u/0/folders/1cKRxE9S8sb23oDlQd008UV9nmzOWXYwX

Web Hi-Fi Prototype Code: https://github.com/Rintel/UCD-HiFi-Prototype

MacOS Hi-Fi Prototype Code: https://github.com/nepysb/SarosUCDMacOSPrototype