# AR Notes

## A groupware/mixed-reality project

Michele Romani, Rusna Rusna

AR Notes is an augmented reality task organiser designed to improve teamwork in a shared coworking space. It features a combination of software technologies and physical interactions on a physical Kanban board to take advantage of the strengths of both the virtual and the real world.

## Brainstorming, Inspiration and First Concept

The initial idea of our project is to create a tool that could take advantage of the physical world to improve the user engagement in daily tasks and their collaboration in a shared environment. We identified different groups that could take advantage of this tool: students, workers, families, and teams.

We also identified different tangibles object that could be used for an intuitive interaction: cards, marbles, cardboard shapes, coloured shapes. These inspirations came from different other ideas below:

1. Marble answering machine concept - Durrell Bishop
2. Virtual Reminders concept – Augmented Realitease

Our first concept was a post-it board in which everyone could attach an augmented post-it for the others to see, easily copied and shared. On top of it, the users could also have the digital version always available on their devices. Our brainstorm report can be found here.

## Interviews & Feedback

We created an interview guide and interviewed participants. The goal was to talk to participants in different user groups - students, those living in a residence and who may use notes to communicate amongst each other and working professionals. We talked to five students, one of which who used notes in their household and one working in an office as PhD. We also reiterated the process with a university professor that gave us a professional point of view.

Some key points emerged that made us realise we needed to reiterate our process and modify our first concept, in particular:

- Post-it gets easily lost, they should be easy to replace
- There are very user friendly apps, such as Google Keep, to take notes and share
- Physical reminders are better in a shared environment, you may "lose" track in a group chat
- Paper on the desk is a better reminder than a notes app

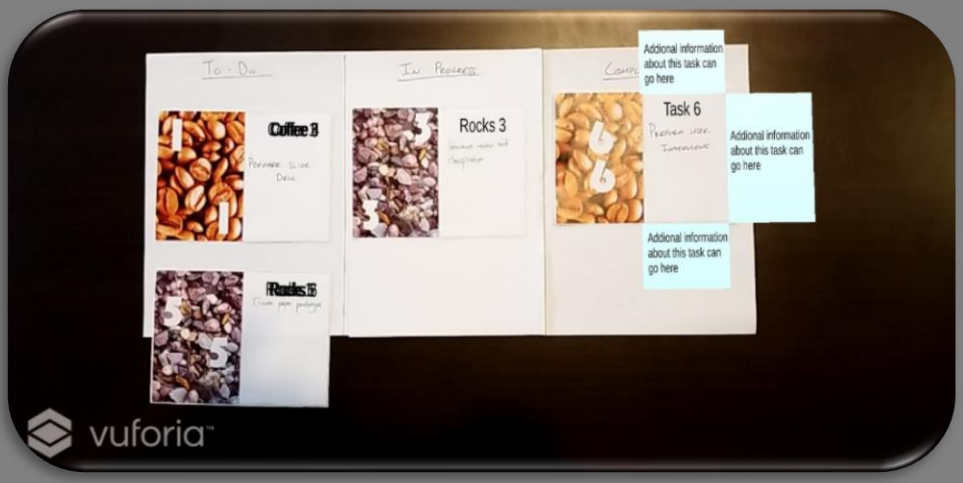
The full interviews documentation can be found here.

## Second Concept

The new concept we came out after reiterating our idea is an AR task board for professional teams, like a Kanban board. The board and the tasks themselves will be physical, and each task on the Kanban board will act as a marker which can be used to display supplementary information about the task in AR. Tasks could be duplicated and shared by using duplicate markers. We experimented other interactivity of the board and markers, like showing task progress and change it accordingly. The complete board can also be accessed digitally on a web interface on any type of device while the AR client can be used on any mobile device equipped with a camera.

## Paper Prototype

We built our first prototype using paper and food images as markers. We used this prototype to test the augmentation with Vuforia development kit and explore possible interactions with the physical cards and between different users.



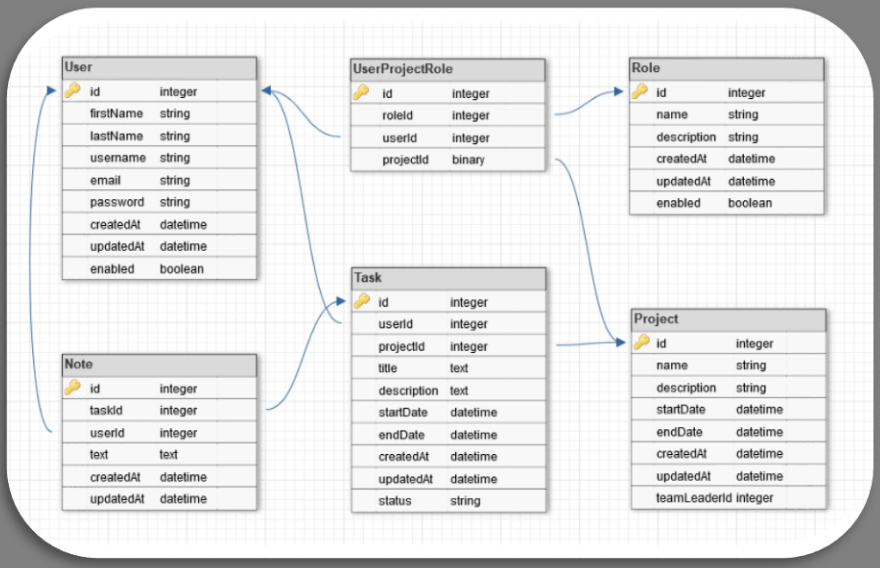

The main concepts we learned from this prototype were:

- The possibility to add extra text/photos/videos to a paper task
- The possibility to have a community version of each card on the board and a duplicate that is given to the task owner. Having the same marker, it means that every printed copy of a card is also a digital copy of its content
- The possibility to use a projector to synchronize the physical board with its digital counterpart.

Sketches and presentation of our first prototype can be found here.

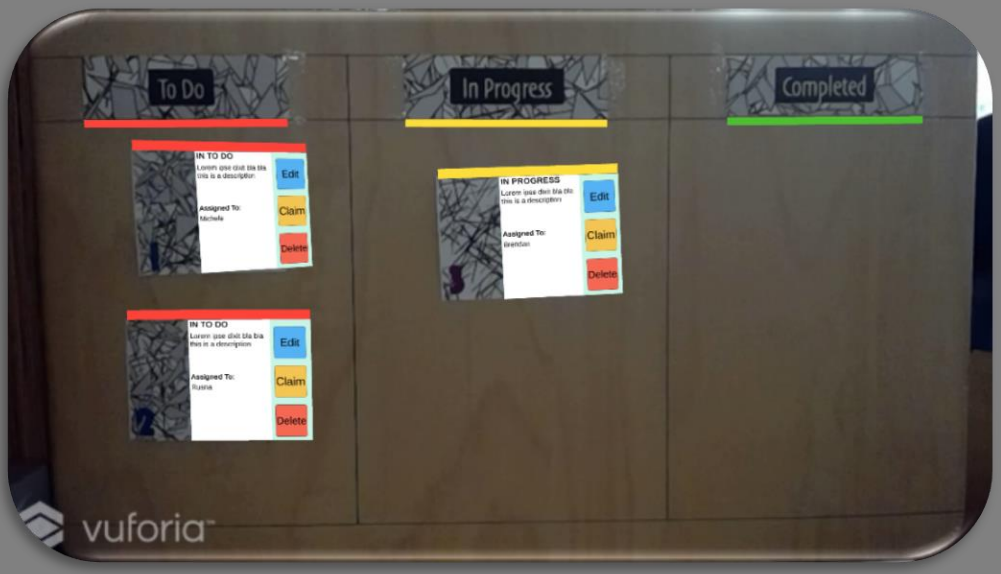
## From paper to…
## a complete software system

Building a complete software system out of a paper concept has not been an easy task. The first step was identifying the entities of our system and how they needed to interact.




We used **dbdesigner** to build an Entity-Relationship schema for our application. We then exported the SQL Create statement and imported it into PostgreSQL.

We also implemented a Java application with RESTful API that manages the layer between clients and the database and we modelled all the entities, both on server and clients, to match the schema.



## System Architecture & Technology Stack

The system is a **distributed virtual environment composed** of 3 different modules that communicate each other with REST calls, it's based on a classic centralized client-server architecture.

- **ARNotes**
  An Android client built in Unity and powered by Vuforia AR library. It's the main interaction point between the users and the application physical objects.
- **ARNotes – Server**
  A Java application that provides all the core logic and functionalities of the backend.
- **ARNotes – Web**
  A web interface for managing the digital version of the application objects.

To build the system we used several technologies, languages and libraries according to our requirements:

- Unity Engine (C#) – Game engine that we use for the AR clients.
- Vuforia – AR library we used to render the augmented tasks.
- Java Spring (Java 13) – Last version of the OOP language we used to build the server.
- Maven – Package manager for Java libraries and dependencies.
- JavaScript EC6 – Last version of the web language we used to manipulate objects on the interface.
- Bootstrap – CSS/JS library for building consistent interfaces.
- PostgreSQL – Database management system.
- Swagger – API viewer we used for testing the application APIs.

## Functionalities

In current prototype the application supports both **co-located** and **remote** collaboration, thanks to the server and the persistency of the database. We tried to keep the virtual environment always **consistent** with the continuous changes that the users may perform.

The **responsiveness** of the system is decent when interacting with a limited number of virtual objects in the same scenario, however the nature of the Kanban board allows us to be flexible on this.

The following functionalities have been fully or partially implemented:

1. CRUD methods for managing entities (Task, User, etc.) through APIs
2. A *claim* method to assign a Task to the current user.
3. An *assign* method to assign a Task to a specific user.
4. A *state recognizer* that changes the state of a Task according to its position on the Kanban board.
5. Real-time editing of the content of a Task
6. Synchronization of the AR Notes clients using the API
7. Synchronization of the digital web board using the API

## Known Issues

The main issues we have with our system are related to **markers recognition** and **synchronization (WYSIAWIS).** We noticed that having different objects in the same close space requires more computational effort and this usually leads the application to a lack of precision or eventually to not being able to identify the object.

Another problem due to the client-server architecture and the physical-digital duality is the synchronization. A distributed system like ours require the clients to keep "polling" the server with requests to update the current state of the system. If the polling happens in a too short interval of time, even with a few clients, the server gets flooded by requests and eventually is unable to answer all of them. An opposite situation happens when the clients do not update their status enough frequently to be updated about changes between them, leading to inconsistency of the contents or worse to overlap changes made by another user.

The main synchronization problem is between the physical and the digital board: when updating the physical board, it is quite simple to update the server and then synchronize the digital board accordingly, but doing the opposite is a hard challenge. In the following section we discuss a possible solution.

## Future Developments

The project has a lot of potential and we are planning on developing further to improve existing functionalities and add new ones. A main functionality we had no time to implement is **user authentication** and content visibility as we want our user to be able to keep some of their notes "private" while sharing others with the team.

For the physical synchronization problem we were thinking to use a combinations of **cameras and a projector** that are always watching the physical board and whenever the state changes a visual notification will appear on the physical Task asking the users to reposition it accordingly to the new state.

We also aim to improve synchronization, maybe using a system or rules, locks and ownership to prevent users from having conflicts while editing the same Task.

Finally we would like to use a magnetic board with magnets instead of paper and wood, that would improve the physical interaction with the board and at the same time make our system easy to setup on any office board.

## References


- A Tangible Interface for Organizing Information Using a Grid
- https://blog.valiantys.com/en/jira-software/sync-physical-agile-board-jira/
- https://www.psoda.com/global/blog/psodavision-sync-physical-and-digital-kanban-boards/