

MILESTONE I

Proof Of Concept

NSO^QRE

INSQUARE

insquare.tk

TEAMWORK

SPLITTING TASKS

How did we manage everything?

- 3 of us on Back-End, 2 on Front-End
- We used Trello to manage our tasks as our mentors suggested
- Task list was priority based: the work has been managed using the **Agile** paradigm

What was easy?

- Writing code
- List User stories
- Message broadcasting
- Elasticsearch APIs

What was difficult?

- Choosing the best platforms
- Understanding NoSQL DBMS
- Connecting Back and Front End
- Authenticating users

Why?

It is hard to approach technologies you've never seen before

THE MAIN FEATURE

SEND A MESSAGE TO A “SQUARE”

Where you go. Who you meet.
Real time location-based reachability

As a logged user
I want to send a message to a square
to solve a local-based need.

People on the go want to be provided with *fresh* information about their **locations of interest.**

OUR PROGRESS

WHAT WE ACCOMPLISHED

Facebook Log-in (As a logged User)

- Node.js (Passport, Express)
- Facebook SDK

Chat on Android (I Want to send a Message)

- Socket.io - Message Broadcasting
- Client-Server Connection (RESTful Api)

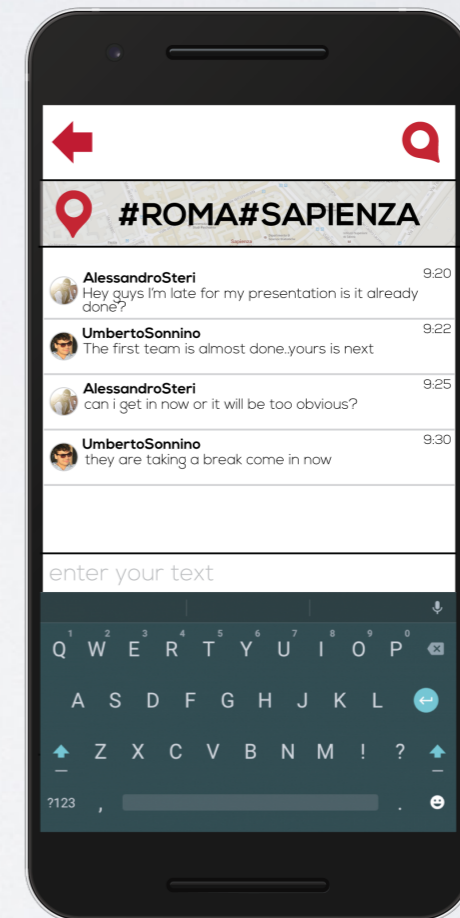
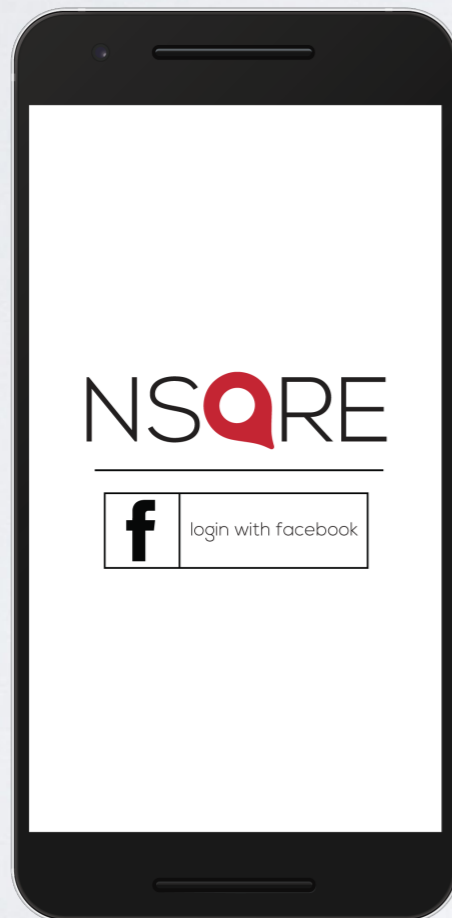
Persistence Data(To solve a local-based need)

- Elasticsearch (Search Engine)
- MongoDB (NoSQL DBMS)

OLD UI

IT STARTED LIKE THIS..

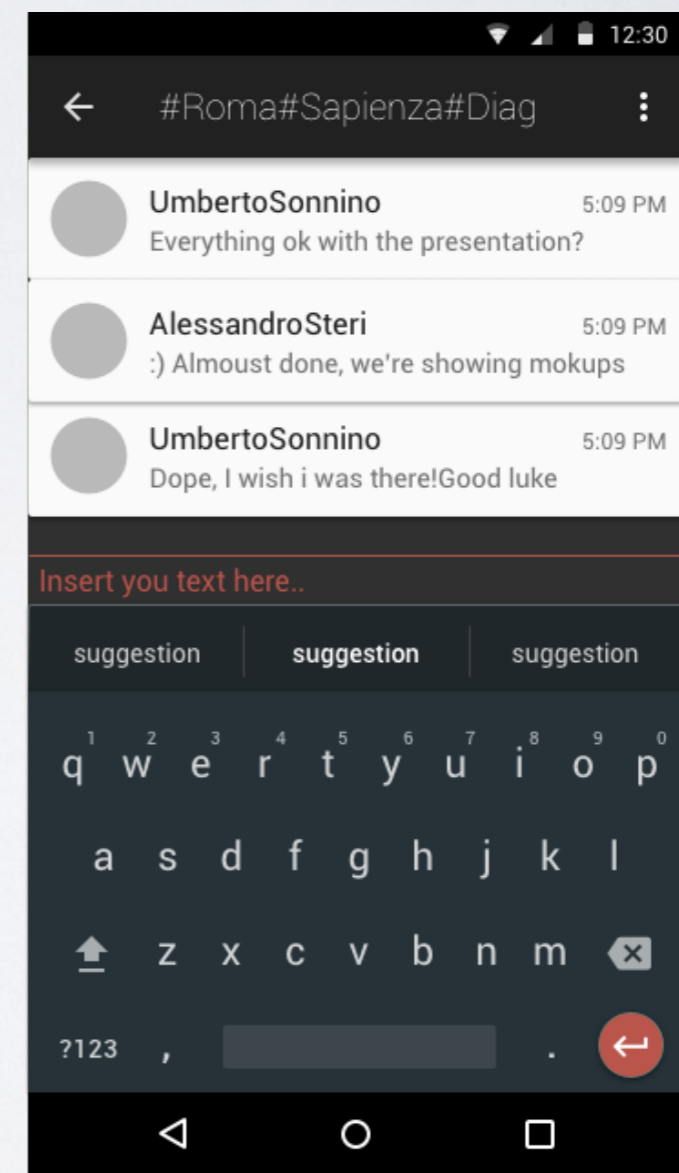
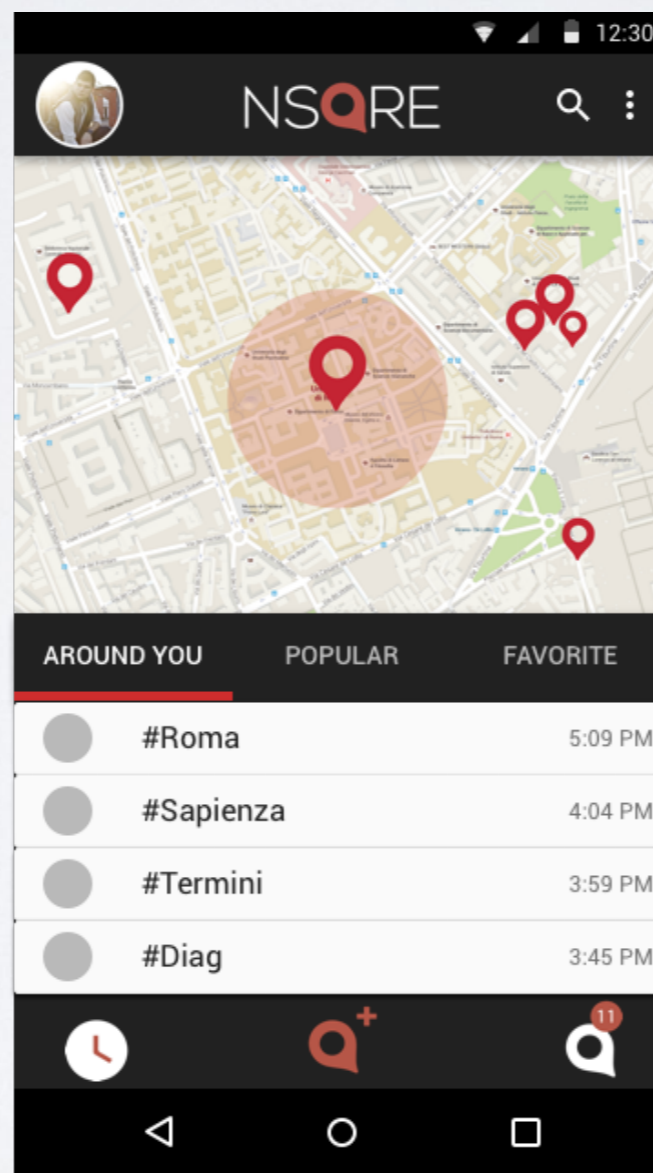
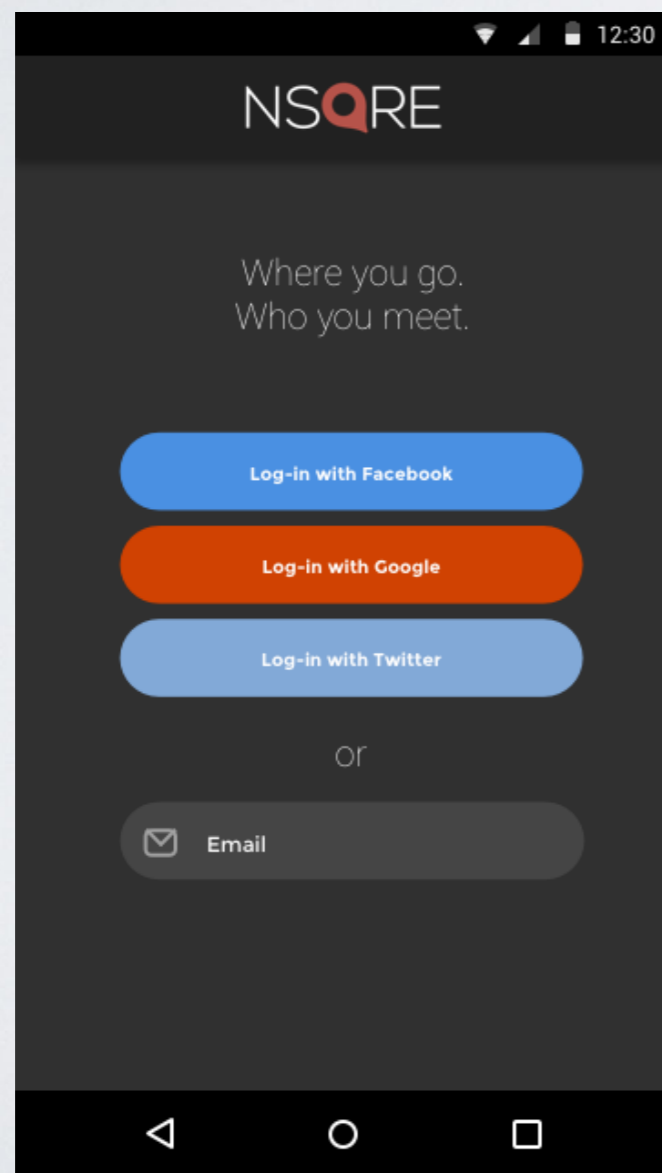
- Made without involving potential users
- Not based on Google's Material Design guidelines
- Lacking features and logic developed during Milestone I



REDISIGNED UI

OUR VISION

- Some potential users have already been involved
- Inspired by Google's Material Design
- Now with a clearer vision on features and logic developed during Milestone 1



OVERALL ARCHITECTURE

ORIGINAL VS ACTUAL ESTIMATION

External Resources:

Google Maps APIs & locations information
Facebook & Google APIs and User database

Google Maps APIs & locations information
Facebook, Google, Twitter APIs and User database

Server Side:

Find a **#Square** around you
Store user's data through SQL Database

Data Persistence and Search (Users, Squares, Messages)
NoSQL Database (ElasticSearch, MongoDB)
Socket.io Message broadcast (Server-Side)

Client Side:

Create a new #Square to share stories
Interactive, maps-based UI
Use hashtags to define a specific location path

Android-App with Interactive, maps-based UI
Create Square through SquareTag and send Messages
Socket.io for realtime communication (Client-Side)

ROAD MAP

GOALS MILESTONED



First Milestone *(Jan 18th)*

- First form of persistence with a Cloud-Based DBMS
- Rudimentary working chat on Android
- Facebook login integration



Second Milestone *(Feb 15th)*

- Google, Twitter, Local login integration
- Full Google Maps API Integration
- Hashtag (SquareTag) tree-based structure
- Improvements of the overall UI/UX with first user pilot



Third Milestone *(Mar 15th)*

- Fully built mobile UI on Android
- Tailored suggestions for nearby #Squares
- Push notifications

Our Milestone list is depending on the user feedback we are going to collect

CONCLUSIONS

- Develop thing and make choice makes the **roadmap more clear**
- Collecting feedback and **involving users** make more and more sense
- The **Agile paradigm** pays off