
Whisper Project - Initial Report

Fernando Villa
Khalid Al-Obaid

Mariya Abdiyeva
Nicolas Cabuli

Oluremi Obolo
Santiago Arrubla

7 February 2017

1 Project Description

1.1 Requirements (Aim and Objectives).

Our aim is to create a distributed chat system. We intend to achieve the aim by covering the following objectives in order of priority.

Priority Level 1 (Compulsory)	Priority Level 2	Priority Level 3
<ul style="list-style-type: none">-Secure connection-Encrypted messages-IOS, Android and Web platforms (at least two platforms)-Contact list-Search for contacts-Signup / sign in by email or Gmail-One to one conversation (text message)	<ul style="list-style-type: none">-Send images-Group chat-Chat bubbles instead of username-Notification of received messages	<ul style="list-style-type: none">-Signup/sign in by Facebook-Able to send other type of media (eg video, voice note, etc)Online status of other users-Acknowledgement of messages received/delivered/read including date time-Ability to use chat off-line

1.2 Strategy and Tentative Timetable.

Our strategy is to develop three client-based platforms (ie Android, iOS and Web) using firebase as a server. The reason for including a third platform is to act as a backup should there be problem with any of the platforms during testing which we are unable to debug at the end. The tentative timetable (Figure 1) presents the main activities required to complete our final product. It is important to state that the timetable is constructed based on an agile software development methodology, Scrum [1], which basically allows us to iterate and make incremental/evolutionary contributions. This methodology is part of our strategy. We will also use different Project Management roles like a Scrum Master (who removes team obstacles and pushes resolutions) and the product owner (who acts as the client voice) to ensure a high-quality product. Further details about the assignment of roles are in Figure 2 and Section 2.

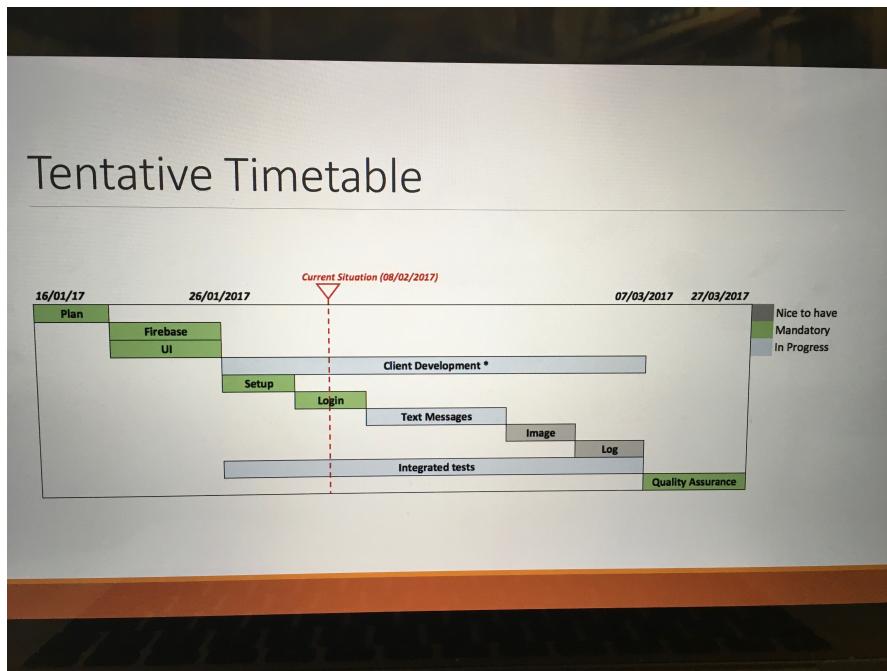


Abbildung 1: This Timetable was uploaded via the project menu.

Figure 1

(*) Three clients development in parallel (Android, IOS and Web)

Our main activities are explained as follows:

Plan: During planning, we defined how we were going to work, our goals and organization as a team. We also discussed our individual strengths and weaknesses in the team and the tools available to us in order to achieve our aim and objectives.

Firebase: For simplicity, costs (free) and security we are using this Backend as a Service platform to have a robust system which takes care of the authentication, rules for accessing the database (read/write) and the storage of documents so that we could focus on the core functionality, i.e. the actual messaging.

UI: Refers to designing the User Interfaces for all the clients.

Client Development: This is the development of the specific application for each type of client (Android, iOS and Web):

- **Setup:** Here we add the required files and initialize the repository on GitHub. Then we create the project on Firebase so that we could share the same API Keys.
- **Login:** Build the login page, authentication using email and password, Gmail and Facebook if possible.

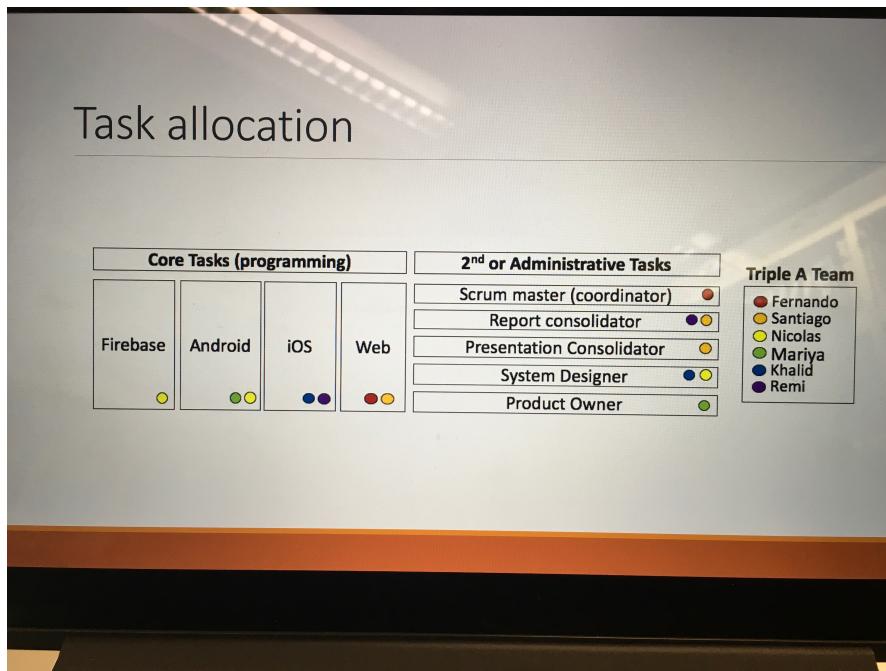


Abbildung 2: This Task allocation was uploaded via the project menu.

Figure 2. Allocation of Tasks

Taking advantage of the previous structure of tasks, we added the role of each member.

- **Conversations:** Text messaging between clients using Firebase as a central database containing the messages between recipients and taking care of the encryption. Add chat bubbles, send and receive messages, show them in an organized manner with timestamps.
- **Image Message:** Share images inside messages (nice to have).

Integration Testing: Test functionality across platforms.

Quality Assurance: Check that all clients communicate with the backend as intended in order to ensure that we cover all the required functionalities and fix any problem that we may have.

Please note that our activities in the client development are organised by hierarchy over time. Thus, developing the text message functionality (which is mandatory) is more important to us than the image message functionality (which is optional but highly desirable). Consequently, we decomposed our activities into tasks and allocated to team members as follows:

1.3 Current State.

Currently, we have developed the authentication functionality inside the three clients. Firebase authorizes the logged users and allows them see the data only if they are logged in.

1.4 Next Steps.

Following our timetable, we are going to focus on making the proper database structure (chats, messages, users, receiver, sender, type of message, etc), rules for accessing the data inside the database (only user x and user y can see their own conversation) and adding new data to the database.

2 Project Organization

In this section, we will give more details about our organization, including how we will make decision and resolve conflicts.

2.1 Roles.

Every team member has two types of tasks viz: core tasks (e.g. programming goals) and administrative tasks (e.g. documentation and update). Additionally, the coordinator is the official team communicator. However, it is clarified that this role does not imply a position of leadership or decision making (the decisions are jointly made by following the decision making and conflict resolution methodology in subsection 2.4). The specific roles were already stated in Figure 2.

2.2 Tools for Communication/Collaboration.

- Slack (for communication purposes and specific tasks).
- GitHub (for code sharing and version control).
- Compulsory weekly meeting every Monday (12:00 PM - 2:00 PM).
- Compulsory Google Hangout every Sunday (5:00 PM- 6:00 PM).
- Extra Google hangout meeting (when needed).
- Meeting on Wednesdays 2:00 PM- 5:00 PM (if needed).
- A timetable and advance curve control (ie where we are vs where we ought to be).
- Our decision-making and conflict resolution methodology (Section 2.4).
- Learning meeting (each 15 day, we arrange a meeting to discuss what we can improve as a team)
- Latex version with template from www.overleaf.com was used to prepare report.

2.3 Peer Assessment Criteria.

Our assessment methodology is based on the following set of criteria:

Each member evaluates others on a scale of 1 and 5 (where 1 is the lowest and 5 is the highest score). For each criterion, each member would write a little justification, which could be replied by the other member. Afterwards, the evaluator could decide to modify his assessment or otherwise. The final score of each member will be the average of all the valuations, weighted by the weight of each criterion. Then, the 100 points would be proportionally distributed to meet the assessment criteria of the module.

2.4 Decision Making, Agreements and Conflict Resolution Methodology.

Making agile agreements and rapidly resolving conflicts are key factors towards maximizing our team output. Our methodology is inspired by the agents and multi-agents theory [2],

Criteria	Description
Amount of work (1/3)	Amount of work that each member contributes. (E.g. difficulty or complexity in the sub-set of requirements achieved by the member).
Team work skills (1/3)	i) communication skills, ii) cooperation (e.g. how a member supports other members) and willingness to resolve conflicts or reach agreements.
Pro-activeness (1/3)	Self-motivation and Pro-activeness to explore and propose new alternatives or options to overcome any difficulty. These include initiative, willingness to learn, good attitude to explore new things, enthusiastic, creativity and innovation.

Tabelle 1: Peer Assessment Criteria

specifically in a practical reasoning agent BDI (Belief – Desires - Intentions).

In this manner, every team member possesses a valid point of view. Any difference between opinions goes through a deliberation process (where personal conflict is a conciliation process, which is a special type of deliberation), that would transform a set of options to a set of intentions (this implies that each time there is a deliberation process, the first step is to state a set of options). Then we will plan and assign resources. However, the deliberation process required a balance due to time restrictions (based on our timetable and curve advance). This implies that we must state a mechanism to accelerate the process wherever it is appropriate (deliberation process output is the ideal, but practically we would not always have time for a full consensus).

Each time we are unable to reach consensus due to time restrictions, we have specified an agreement protocol that precipitates a final decision where the group will move forward. For this, we use the Borda Count vote procedure, to try to maximize the preferences of the entire group.

Finally, for a personal conflict (such as a communication problem), there will be an arbitration protocol to resolve it. Basically, the team identifies those members that are not involved in the conflict and implore them to mediate. If after a conciliation process there is no agreement, referees will make the decision (only if there are 2 or more referees). If a referee abstains, then we would activate the agreement protocol.

References

- [1] Cohn M. Succeeding with Agile-Software Development Using Scrum. Pearson Education: Boston, 2009.
- [2] M. Wooldridge, An Introduction to Multi Agent Systems. Department of Computer Science, University of Liverpool. John Wiley and Sons Ltd, 2009