Team, Vision and Initial Deployment

Software Engineering - Milestone 1

1. Team

For the group project of the discipline of Software Engineering of the MSc. in Informatics Engineering of the Department of Electronics, Telecommunications and Informatics of the University of Aveiro, the team members of the group P54 are: André Pedrosa, Filipe Pires, Helder Paraense and João Alegria.
The work will be distributed equally, while considering background knowledge of each person. João is assigned as the Team Leader, André as the Infrastructure Manager, Filipe as the Documentation Manager and Helder as the Quality Assurance Manager.

2. Project Vision

Group activities such as the famous Trivial Pursuit and other games not requiring boards or cards are a very common way of spending quality time with close family and friends. These activities build rich traditions and valuable memories and stimulate our brains to cooperate and compete in a healthy way. However, they lack the ability to bring closer people who are physically far from each other.

The concept behind our project is simple: we intend to develop a prototype capable of bridging this gap by allowing multiple players to interact with each other at a distance in the game that we intend to offer – the Charades.
The rules of Charades are: one person must choose an idea to act out (this can be something like a book or a movie, or anything else); this person must then try to act out the idea for the group so that they successfully guess what the person was trying to act out; the person who gets it right first gets to have the next turn acting something out.
To do this, we aim to provide an online system capable of streaming the individual acts of one player to all the others within a given session. But there is a twist. In order to "gamify" the concept, instead of simply streaming the live video captured by the players' devices, we will stream a customizable avatar of themselves with the help of Orbbec (a Kinect-like device) that will replicate all the moves made by users. This way, players will not only be able to play with their dear friends but will also practice with other players while maintaining their anonymity.

Regarding the technical accomplishment of such vision, our plan is to develop a solution based on Jakarta (formerly known as JavaEE), providing a web interface for users, persisting data in a PostgreSQL database and interacting with the client-side through a REST API. The data streams for game sessions will be dealt with Kafka. The system components shall be deployed in Docker containers and such deployment shall be automated with the help of Jenkins. During development, tests will be created in various levels to ensure the correct functioning of the entire system. Also, in all aspects of the project documentation will be a great concern, focusing mostly on the public API (possibly with the use of Swagger) and on the code itself.

3. Project Planning

Having established a good notion of what is to be developed for the purposes of the course, it is recommended to define some key aspects before proceeding to the actual implementation and code writing. With this in mind, here we present the definition of our target audience, the possible usage scenarios, the functional and non-functional requirements and the resulting main features of the system.

3.1. Target Audience

In order to make a solid representation of our target audience, we resorted to the method of *personas*. A persona is a fictitious person that serves as an archetypical user of a system, an example of the kind of person who would interact with it. In our case, we created three complementary personas to more closely define our target audience. These are presented next.

Frances Miller

Sixty Seven-year-old Frances is the mother of four children. She lives in her own home, bakes a pie once a week so that she has something to serve for Sunday visitors (usually her children and their immediate family). Mrs. Miller likes to spend her time in traditional non-technological activities, but she is familiarized with the basic usage of computers and other devices like smartphones.

Frances was thrilled with our idea as she misses playing with her children on the weekends which they are not able to visit her. Her main interest would be to enter gaming sessions with them and occasionally play, but she would like it to be as straightforward as possible.

Jack Miller

Twenty Six-year-old Jack, on the other hand, is the recently married husband of Frances' youngest daughter. Although not much of a fan of physical activities, he is very interested in the idea of pleasing his mother-in-law by making these long-distance charades games work.

As a long-time gamer, he is very comfortable around computers and new software and is actually liking the idea of playing with strangers before doing silly movements around Frances.

Peter Lafront

Fourteen-year-old Peter is a young teenager with no siblings that lives with his parents far from the Miller family. He loves to play any kind of games but suffers from partial physical impairment in his legs.

His parents thought that our system could be a good way for Peter to exercise his muscles indoors while not feeling so lonely nor ashamed with his movement difficulties.

Peter is open to such experience and was happy about the idea of customizing his own personal avatar.

3.2. Usage Scenarios

With the personas defined, we can proceed to studying the most probable and appropriate use cases. At this stage, the scenarios have a high-level nature and are open to updates. The personas are here used to easily present these usage scenarios:

Scenario 1 – Registration

Jack creates an account for him, his wife and his mother-in-law so that they can play together.

Scenario 2 – Authentication

Frances logs in with her credentials so that Jack can identify her when creating a session.

Scenario 3 – Private Session Creation

Jack creates a game session open only to players invited by him.

Scenario 4 – Session Configuration

Jack decides the rules of the session (such as how the actors are chosen, how many points are needed for a player to win, how many rounds, etc.) through the configuration panel.

Scenario 5 – Session Invitation

Jack invites his wife, her siblings and his mother-in-law through their usernames.

Scenario 6 – Invitation Acceptance

Frances accepts the invitation received by Jack and joins the session created by him.

Scenario 7 – Public Session Creation

Peter creates a game session open to anyone.

Scenario 8 – Public Session Listing

Jack checks the list of currently open sessions (created by friends or anonymous users) so that he can practice before playing with his mother-in-law.

Scenario 9 – Public Session Entrance

Jack enters Peter's game session and waits for the game to start.

Scenario 10 – Avatar Choice

Peter customizes an avatar that will represent him in game sessions, choosing between options such as color of the avatar (and perhaps height or event 3D model).

Scenario 11 – Game Session

Jack starts the game and the Millers take turns being the actors that perform based on the charades they are given, and the participants who are guessing gain points when they find the right solution ahead of the others.

Scenario 12 – Spectator Mode

Frances is tired of playing, but still wants to be in the game session and interact with her family, so she decides to change her status from player to spectator, so she can watch her family play and still have fun by spending time with them.

Scenario 13 – Exiting Session

Peter plays for a while in an open session but decides to search for a more crowded one, so he exits the current session he is in and goes back to the list of open sessions.

Scenario 14 – Automatic Ending Session

The system ends the session according to the rules (e.g. all rounds have been completed) and presents the results to all session players, as well as the winner.

Scenario 15 – Ending Session

It's late, so Jack and his family decide to stop playing, and Jack ends the game session they are in. With that, the session results and the winner are shown to all player.

Scenario 16 – Closing Session

Jack closes (deletes) session and all players are duly notified.

Scenario 17 – In-Game Answers

Peter is playing in a session and watching another participant perform a charade. He thinks he knows the answer, so he types it and submits. All participants are notified if the answer is correct, which ends the turn of the actor and awards a point to Peter, or if it is incorrect, in which case the turn keeps going and Peter may try another guess if the rules of the game session allow it.

3.3. Solution Requirements

We now move on to the functional and non-functional requirements of the system. Consider functional requirements as the descriptions of ways the product must behave, whereas non-functional requirements, also known as quality attributes, are descriptions of the general software characteristics.

These requirements are presented in the table below.

Functional (The solution must…)	**Non-Functional**
Be able to accept registrations and account deletions.	Registration and authentication must be seamless to users.
Be able to authenticate already registered users.	Web interface must be very simple and intuitive.
Store and make available the basic information of each user.	User information must be quickly accessible.
Allow the hosting (creation) of game sessions.	Requests / invitations / notifications must not take longer than 5 seconds to be transmitted.
Allow the invitation / acceptance by a host of other users to a session.	Open sessions listing must not take longer than 3 seconds to refresh.
Allow the listing of open sessions.	Avatar movement streaming must not have a delay longer than 1 second.
Allow requests for entering sessions.	Results presentation must be done in less than 5 seconds.
Allow the definition of a charade solution.	
Allow the starting/ending of a game by the host.	
Automatically recognize input devices.	
Define users without camera as passive session players.	
Accept attempts / suggestions by session players.	
Detect when a player solves the charade.	
Notify players when the game ends.	
Define the actor of the current round (either random or the winner of the previous round).	
Present the session results at the end of each round.	

Table 1: Solution requirements of our system.

3.4. Main Features

Finally, we have everything we need to define the main features to be developed. The most important features that are paramount for the success of this project are listed below:

- **Charade Session:** the core of the project, a Charade Session is a room in which people take turns being actors (the ones who receive a theme and must perform to the camera in order to give hints to the charade solution) and participants (the ones who take guesses while watching a video of an avatar mimicking the current actor's movements);
    - **Private Session:** a session can be private, as in only friends or people invited by the host player can participate or watch;
    - **Public Session:** a session can be public, in which anyone can request to enter the session and play or watch others play (as spectators), so it doesn't need invitation nor do the players need to be friends.
- **Avatar Creation:** for anonymity, shyness, or just to have fun, all players have Avatars (models that can be customized and are shown to others when a player is acting a charade) and have a menu where they can create or customize them;
- **Seamless Interaction:** players can play a game of charade and interact with each other when in a session, having an Avatar representing them with a fast transmission and reliable GUI that is easy to learn and use;
- **Streaming of Avatar:** in a Charade Session, all movements done in front of the camera are captured by it and replicated by the server in a video, represented by the actor's Avatar, that it streams to all participants in the session so they can watch the movements and take their guesses;
- **Hosting of Sessions:** any user can host a session of the game and decide the ruleset, if the session is Private or Public, and choose to invite friends. The host can decide when the game ends, and even remove players from the session if needed;
- **List of Sessions:** all users will have access to a list that shows all public sessions that they can join, and even private sessions from their friends and family, to which they can request entry and await acceptance by the host of the private session.