

## Department of Languages and computer's science

Cloud Application Development

### Case study: Treasure Hunt

In this case study you must develop a web application for a *Treasure Hunt* and deploy it over a *PaaS cloud*. The *Treasure Hunt* is a competition game in which a group of participants

They must find a series of treasures scattered throughout a play area. The location of each treasure will come indicated by a clue that suggests where it is hidden. The first winner will win the game

Locate all the treasures.

The following is an informal list of the main requirements of the application:

- **User identification**. Some external secure identification system will be used, for example OAuth 2.0 or the *Google* user API, so that the connection to the application and interactions with it they are made authenticated from the email of the player. The application will allow  
The players *login* and *logout*.
- **Relationship of games**. The application will show a list of the treasure hunts registered in the system, indicating those that are active (those in which no participant has located even all the treasures), or who has been the winner of those that have already finished.
- **Game creation**. Users will be able to create new instances of treasure hunts, defining the play area (rectangular, identified by the GPS coordinates of its center and its dimensions), the number of treasures to be found, their location, and the clue that leads to each of them (a text and / or an image alluding to it). Only the organizer of a game can access this data.
- **Supervision of games**. The organizer of a game can monitor the status of the game, displaying on a map the play area with the location of the different treasures, together with the relation of participants who found each of them, and the state of the game (if there is already a winner or not). He  
The organizer will also have a button to reset the game, eliminating all  
Treasure find data recorded so far.
- **Participation in a game**. Users can decide to participate in any of the fighters that are active at any given time, so they will have access to a map of the playing area as well as to the clues to find the corresponding treasures. Users may indicate that they have located a treasure, then showing it on the map. As proof of the finding, they will rise to the system an image of the treasure found. When a player locates all the treasures, he will be  
You must report that you are the winner. From that moment on, no new findings of treasures for this game, indicating to the rest of the users that the hunt has ended and who is the winner when they try to do it.
- **Administration**. To facilitate the administration of the system, it is convenient that there are users with Administrator role and ability to access all created games. The user [testparaingweb@gmail.com](mailto:testparaingweb@gmail.com) will have administrator privileges.
- **Interaction between users and social networks**. The application will allow interaction between users and the use of social networks, allowing for example:  
or the sending of messages between users (so that the participants of a hunt can collaborate between them, so that the organizer of a game can provide additional clues to any of the players, etc.) or integration with social networks (for example, Twitter or Instagram), so that the organizer of a game announce the availability of it, or to notify who has been the winner.

- **Visualization of maps in images.** As indicated, the game area of the fighters is will show on a map (for example, OpenStreetMaps or Google Maps) where markers will be used to indicate the situation of the treasures (those found in the case of the participants of a game, or all in the case of the game organizer or application administrators). The system

It will also manage the storage and display of images (for example, relating to treasure clues, evidence that they have been located by players, etc.)

- **Geolocation and geocoding.** The system will use geolocation to show on the map the position of a participant (for example, using Google Awareness), as well as geocoding for that the organizer of a game can facilitate the location of the treasures (by indicating their address postal or placing the markers directly on the map) from which the GPS coordinates of the treasure. To avoid saturation of the free layer of servers geocoding will be implemented some *caching* system that avoid the repeated call to the same.

- **Storage** Information storage games (play area, location of treasures and clues, images, treasures found by each participant, etc.) will be performed on a basis of non-SQL data, for example MongoDB, Cloud Firestore or the Google App Engine Datastore.

- **Deployment.** The application will be deployed in a PaaS cloud, for example AppEngine or Heroku.

### Delivery

The case study will be delivered through the virtual campus, through a compressed file containing:

a. the complete code of the developed application.

b. a report indicating the technologies used (programming language, *frameworks* , etc.),

a technical description of the web application developed and the design of the database, and the

installation instructions (if necessary), and also describe the functionality

implemented, identify through its URL the external APIs used, and finally indicate

Possible limitations of the proposed solution and problems encountered during its development.

c. the memory must indicate the **URL** (<https://IDProyecto.appspot.com>) where the application is running

in *Google App Engine* , and the code and name of the developer group participants.