Albiziapp: a Web, Collaborative and Gamified Tool Dedicated to Tree mapping and learning

Pierre-Yves Gicquel, Ludovic Hamon, Florian Plaud, Sébastien George

Albiziapp: a Gamified Tool Dedicated to Tree Mapping

Pierre-Yves GICQUEL[1], Ludovic HAMON[1], Florian PLAUT[1], and Sébastien GEORGE[1]

LIUM - EA 4023, Le Mans University, 72085 Le Mans, Cedex 9, FRANCE
{pierre-yves.gicquel, ludovic.hamon, florian.plaut.etu, sebastien.george}@univ-lemans.fr

Abstract. Identification and mapping of trees through a Geo Collaborative Inventory (GCI) platform is an important task for research in botany, citizen sciences and education. The quantity and veracity of the recorded data rely mainly on the motivation and engagement of each participant. However, for a non-botanist, tree mapping can be perceived as an unstructured and tedious task that requires advanced skills. In addition, existing GCI applications deliver poor or nonexistent feedbacks regarding identification and mapping skills as well as progression in these skills. Structuring GCI sessions and enhancing them with game mechanics and clear objectives may have a positive effect on the inventory quality. Inventory situations being highly context dependant, a descriptive model of the GCI task is required to create adapted activity scenarios usable in various situations. This paper presents Albiziapp: a web collaborative and mobile tool, based on OpenSteetMap, that operationalizes any gamified scenario, consistent with a descriptive model built from a structural analysis of the GCI task.

Keywords: Tree mapping & learning · collaborative inventory · gamification

1 Introduction

Geo-locating trees and learning to identify them is especially interesting for domains such as botany, citizen sciences and education [6, 13, 14]. One can use a mobile device with a Geo-Collaborative Inventory (GCI) application to create Point Of Interests (POI) representing trees together with information (*e.g.* name, genus, species, size). Getting relevant and regular Volunteered Geographic Information (VGI) relies mainly on the motivation, engagement and skills of participants. However, despite the existence of numerous web or native GCI applications, such as OpenStreetMap (OSM)[1] or Spotteron [2], most of them suffer from several issues linked to: (a) a non-adapted interaction paradigm to record new trees in a short time (b), the absence of structures leading the users in the

[1] https://www.openstreetmap.org
[2] https://www.spotteron.net/

task and (c) appropriate mechanisms enhancing their motivation, engagement and identification skills [11].

In the gamification domain, this can be counterbalanced by structuring the inventory process through an adapted scenario. This scenario must define: (i) clear objectives, (ii) possible user actions (iii), immediate and appropriate feedbacks for these actions and (iv), clear progress indicators in terms of tree recognition and mapping skills [5]. The learning and inventory situations being various (*e.g.* residential areas, natural parks dedicated to some species), a descriptive model is required from which an adapted inventory and a gamified scenario can be formalized and tested.

This paper presents Albiziapp[3], a web application dedicated to the operationalization of gamified scenarios for the collaborative inventory of trees at a large scale. The main contribution of this work lies in the design of a descriptive model formalizing gamified scenarios for collaborative inventories. This model aims to tackle the following challenge: to allow everyone to write a gamified scenario that is operationalized by Albiziapp. The player will follow a set of activities made of clear objectives (*e.g.* add, identify, check a number of trees), actions (*e.g.* give the genus/species, confirm/disconfirm data, take a photo) and, instant or short-delayed rewards related to game mechanics (*e.g.* scores, timer, trophies, status).

In the next section, the GCI related work is studied from the tree inventory perspective. Section 3 proposes the descriptive model for the formalization of a structured and gamified scenario. The Abliziapp architecture and main functionalities are described in section 4. A discussion around the model and the current architecture is made in section 5. Section 6 ends this work with its perspectives.

2 Related Work

For mapping and sharing geo-located objects, OSM is one of the most popular and efficient web services from which, several web or native mobile applications were built, such as: Osmand, OSMContributor, StreetComplete, GoMaps, Vespuchi, MapContrib[4].

Most of these applications relies on the visualization, creation and edition of POIs. In the context of tree mapping, the users can choose and fill a form dedicated to trees or create their own fields for the species, the genus, the common name, etc., of a tree. Searching the appropriate tree form and/or creating the missing required fields for each POI is time consuming for one only interested in plants and trees inventories. With MapContrib, a layer specializing the inventory for a specific category of POI can be created. Once a layer is activated, each new POI will automatically display the appropriate form made of the same set of fields to fill. Nevertheless, it is not possible to edit OSM POIs, while the reading and editing of the user's POIs can only be done after an expert validation.

[3] https://pre-prod.albiziapp.reveries-project.fr/
[4] https://www.mapcontrib.xyz

More ergonomic applications for gathering knowledge about trees and plants can be found in the crowdsourcing domain. Nguyen *et al.* reviewed the systems dedicated to botanical data collection and their automatic identification tools [11]. Various efficient applications are associated with a botanical database for automatic or semi-automatic plant identification such as Pl@ntNet [11], CLOVER [10], Flower Checker[5], MyGarden Answers[6], Garden Compass[7], Garden Tag[8]. Yet, one well-known issue of crowdsourcing pointed by Misra *et al.* is "Recruiting and retaining the participant" [11].

Two other well-known issues in GCI can be noted: (a) the uneven geographic distribution of new POIs and (b), the uneven ratio between POIs creation and edition [2]. To counterbalance these issues, Antoniou *et al.* built a simulation of a gamified version of OSM with virtual players following the principles of geogames [2]. Geogames are games where the navigation task in the real world is at the core of the game scenario [7]. Despite their interesting "theoretical" results in terms of motivation and engagement, they did not simulate the user's interests for the POI categories. In addition, to our knowledge, few works addressed the recurrent problems of GCI through geogames and none were specialized in trees. Dhuny *et al.* built a web mobile system, based on an HTML5 framework, to geo-tag various fruit trees, medicinal herbs and share their location in a public database [4]. This initiative aimed to encourage planting for feeding purposes. Each POI could contain usual data about the plant with fructification and flowering information. A badge mechanism was implemented, despite its unclear aspect in terms of feedback and customizable features. Finally, the proposed authoring aspect was also fuzzy and seems to rely only on the framework itself.

The presented work is conducted as part of the ReVeRies project[9] and is supported by the French National Agency for Research (ANR). This work contributes to a research axe led by the following hypothesis: a gamified activity of geo-collaborative inventory of trees can enhance: (a) data quantity and veracity and (b), motivation and engagement of the participants. However, given the diversity of inventory situations, GCI scenarios and their game mechanics must be adaptable. This adaptation process implies to: (i) analyze and structure this activity in terms of relevant action units (ii), develop a model describing gamified scenarios. The next section presents an analysis of the tree-based GCI activity, followed by a proposition of such a descriptive model.

3 Toward a Descriptive Model of Gamified Scenario

3.1 Requirements for a GCI Activity

The minimal set of information expected by botanists for tree identification seems to be the GPS coordinates, the species and a picture [11]. For a GCI,

[5] http://www.owerchecker.com/
[6] http://www.gardenanswers.com/
[7] https://www.gardencompass.com/
[8] https://gardentags.com/
[9] http://reveries-project.fr

each of these pieces of information being a valid contribution, a POI with at least one of those fields is acceptable. The enhancement of the record veracity and completeness by the community is a relevant problematic for citizen sciences [2]. As a consequence, the possibility for everyone to make new records and/or edit them must be taken into account. However, records verification is a tedious task that is mostly done by experts. Tools for the automatic identification of plants from leaves can also be used with VGI systems such as Pl@ntNet [11] or, in the ReVeRies project, FOLIA [3]. Therefore, actions that "encourage" the data verification (*e.g.* the use of FOLIA, consulting the record history to edit it, records confirmation/disapproval, etc.), must be available and promoted. Finally, to motivate and engage the users, their progress in terms of tree identification and inventory skills must be characterized through appropriate feedbacks. Depending on the inventory and learning situation, several contextual reward mechanisms can be applied based on all or a part of the completion of the scenario and/or the user's actions.

3.2 Descriptive Model of a Scenario for GCI

Figure 1 proposes a descriptive model for GCI scenarios. A record is a set of data containing, at least, the GPS coordinates of the tree. The genus, species, common name and photos, can be added. To create scenarios, we modeled the concept of "mission" as a set of "activities" (Figure 1, black part, *cf.* "ActivityList" class). An activity (*cf.* "Activity" class) is a task to perform that can have three "Types" (in bracket): (a) make X new records [INVENTORY] (b), check and/or edit X records of another participant [VERIFY] and (c), identify X trees [IDENTIFY].

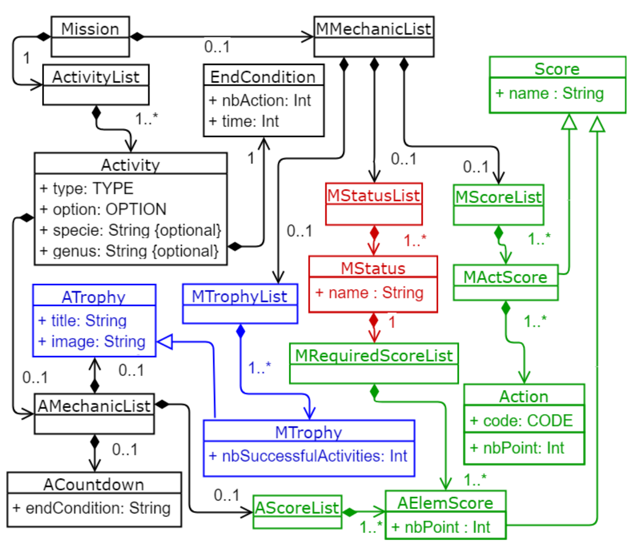

Fig. 1. UML diagram of the descriptive model

This last activity aims to test the identification skills of the user (*cf.* section 5 for more details). An expert geo-tags some trees in the field beforehand. Then, the user has to go to the tree location and fill its data that will be compared with those of the expert. Points are given to the user for each correct answer (*cf.* section 3.3). Each of those types or goals can be refined with an "Option" (*cf.* "Activity" class), asking to perform the activity: (i) on a specified species or genus [SPECIES/GENUS] (ii), with different species or genus [DIFFERENT SPECIES/GENUS] and (iii), freely regarding the tree nature [NONE]. If the "Option" value is either [SPECIES] (or [GENUS]), the "specie" (or "genus") member must be added to the "Activity" object and its value specified. Besides, an activity requires an end condition (*cf.* "EndCondition" class) that can be the number X of records ("nbAction" field), a limited time (*cf.* section 3.3) or both.

3.3 Descriptive Model of Game Mechanics

According to Zichermann *et al.* and Haaranen *et al.*, cited by [4], reward mechanisms are simple, understandable and powerful mechanics to motivate and engage the user . A scenario can be enhanced with two kinds of game mechanics: (i) those associated at the mission level (*cf.* "MMechanicList" class) depending only on the user's actions and (ii), those associated at the activity level (*cf.* "AMechanicList" class) depending on the current activity.

Mission mechanics are made of trophies, scores and status (Figure 4, right). First of all, the user can receive trophies (*i.e.* a name and an image, *cf.* "MTrophy" class) for successfully completing a specified number of activities.

A set of scores can also be defined (*cf.* "MScoreList" class). Being composed of a name (*cf.* "Score" class) and points, each score rewards the user each time a specific action is performed (*cf.* "MActScore" classe). Referring by a "code", an action can be ("code" value in bracket): create a POI [GPS], fill/edit the genus/species/common name field [complete/edit GENUS/SPECIES/COMMON], take a picture [photograph], confirm the record data [validate], doubt the tree existence [question] or use FOLIA [useFOLIA]. Each coded action can be performed at any time and contribute to the GCI regardless the activity type. However, only the appropriate actions can move the activity forward.

A status consists in one "name" or title (*cf.* "MStatusList" and "MStatus" classes) given to the user once a threshold is reached. This threshold is defined by a set of values of one or several previously defined scores (*cf.* "MRequiredScores" classes). Once a new status is obtained, it replaces the previous obtained one (*cf.* Figure 3, right).

Activity mechanics are made of: a countdown to limit the activity in time (*cf.* "ACountdown" class), trophies and scores for successfully completing the targeted activity (*cf.* "ATrophy" and "AElemScore" class).

Note that the "code" vocabulary contains also the keywords [identifiedGENUS], [identifiedSPECIES] and [identifiedCOMMON] to set up a score, giving points for a correct answer on the genus, the species and/or the common name during an [INVENTORY] activity (*cf.* section 3.2). Finally, all the game mechanics are optional as a GCI scenario can also be defined without them.

3.4 A JSON Example

The descriptive model allows formalizing a gamified scenario according to a JSON structure. Figure 2 shows an example of such a structure compatible with Albiziapp. The left and middle-up sides show the mission activities. The first activity aims to find and map 5 new trees of the ”Acer” genus. The second one aims to identify 3 trees mapped by an expert (*cf.* section 3.3) and give 20 ”knowledgePoints” once this activity is done, whereas the last one aims to map as many trees as possible in less than 15 minutes. The middle-down and right sides represent the mission mechanics. A score named ”explorationPoints” give 2 points for each ”photograph”. The trophy ”3 activites done” is given to the user once 3 activities are completed. The user will receive the status ”Traveler” after reaching 64 ”explorationPoints”.

```
"activityList":[
 {
  "type":"INVENTORY",
  "option":"GENUS",
  "genus":"Acer",
  "endCondition":[
   {
    "nbAction":5
   }]
 },
 {
  "type":"IDENTIFY",
  "option":"NONE",
  "AMechanicList":[
   {
    "AScoreList":[
     {
      "name":"knowledgePoints",
      "nbPoint":20
     }]
   }],
  "endCondition":[
   {
    "nbAction":3
   }]
 },
```

```
 {
  "type":"INVENTORY",
  "object":"NONE",
  "AMechanicList":[
   {
    "ACountdown":{
      "endCondition":"time"
    }
   }],
  "endCondition":[
   {
    "time":900
   }]
 }]
```

```
"MMechanicList":[
 {
  "MScoreList":[
   {
    "name":"explorationPoints",
    "actions":[
     {
      "code":"photograph",
      "nbPoint":2
     }]
   }]
 },
```

```
 {
  "MTrophyList":[
   {
    "title":"3 activities done",
    "image":"trophy4.jpg",
    "nbSuccessfulActivities":3
   }]
 },
 {
  "MStatusList":[
   {
    "name":"Traveler",
    "MRequiredScoreList":[
     {
      "name":"explorationPoints",
      "nbPoint":64
     }]
   }]
 }],
```

Fig. 2. A JSON mission example: three activities with a score and a countdown mechanic (left and middle-up) and 3 mission mechanics (middle-down and right).

The next section details the key features of Albiziapp, its architectural and design principles for the implementation of these features.

4 Albiziapp Architecture and Main Functionalities

4.1 A Full-Web, Mobile, Real-Time Collaborative Application

Figure 3 (left) gives a high-level representation of the server architecture and the data flow. Albiziapp[3] is a full-web application, using only standardized web technologies without relying on a specific platform. It has been designed to be used at a large scale. Any mobile or desktop device with a web-browser can run

Albiziapp. Nonetheless, performances and user experience of a web application are known to not be as good as native ones [1]. We detail in this section, how we mitigate these issues.

Figure 3 (left) gives a high-level representation of the server architecture and the data flow. There are, at least two main keys and user-related features directing the back-end architecture (server): (i) recording new trees in a persistent database and (ii), visualizing and editing other participants' records and her/his own in real time.

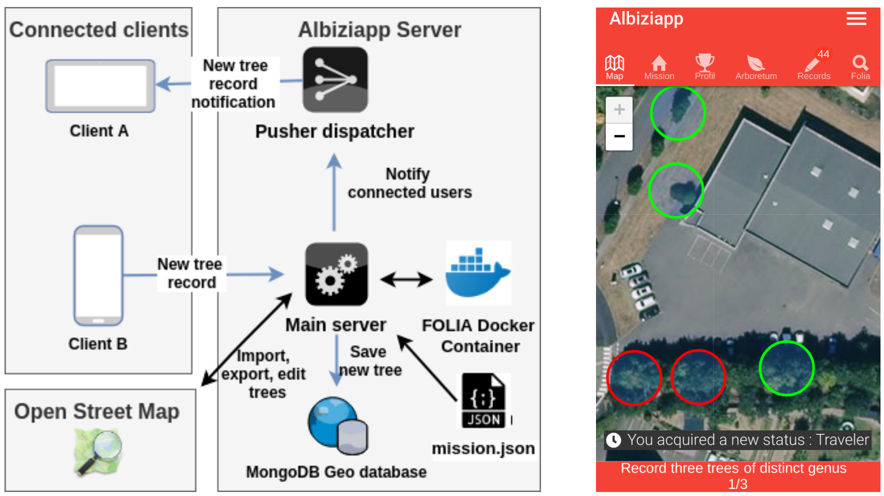

Fig. 3. High level server architecture (left), Albiziapp map page (right)

The first feature entails constraints on the database schema. For a record, there are innumerable types of information that could be relevant when one inventories trees[10], for instance: phenological information (*e.g.* date of flowering), health information (*e.g.* parasite, disease). Various use-cases are likely to arise. Therefore, the database must be able to handle: (a) modifications in the data structure and (b), the coexistence of different structures. Consequently, the NoSQL paradigm was chosen with a document-oriented NoSQL database [9]: MongoDB[11].

The second feature entails real-time communication between connected users. When a user adds a tree, the tree must show up on other participants' map. Such real-time communication is efficiently performed with *websockets* for broadcasting [12]: a client notifies the server that he added a tree, the server then broadcasts the message to all other clients. Therefore, *websockets* were used to communicate.

[10] https://wiki.openstreetmap.org/wiki/Tag:natural%3Dtree
[11] https://www.mongodb.com/

4.2 Interacting with Albiziapp: the Client Side

The web application was designed to be similar to a native application, reusing well-known good practices in terms of interaction[12]. The main interactions are navigation through the different pages and creation/editing/deleting tree records. The navigation follows two usual patterns in mobile applications: (a) the main pages are always accessible through a tab bar and (b), a subpage follows the stack pattern by overlaying its parent (*i.e.* the subpage is added to the stack for access, the page is removed from the stack when leaving).

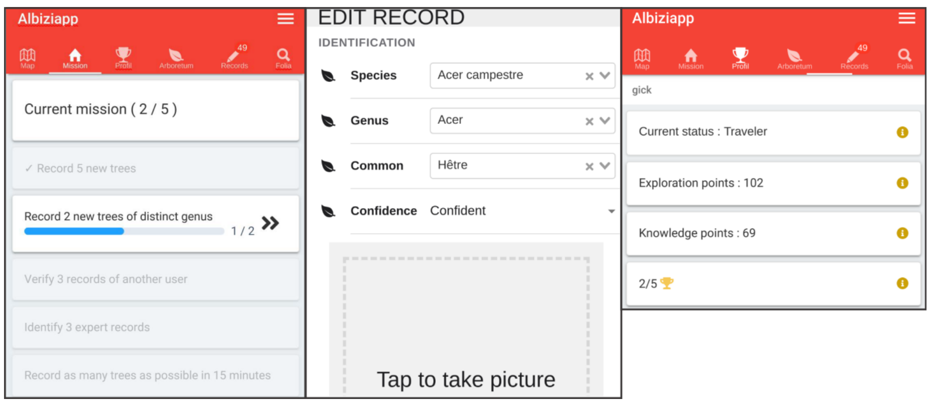

Fig. 4. The mission page (left), the record page (middle) and the profile page (right).

The JSON scenario can be loaded thanks to an option in the main menu protected by an administrator password. Then the mission page is presented with the set of successive activities that must be completed (Figure 4, left). The user can also give up an activity to go to the next one. Interactions involving tree records are initiated from the map page (Figure 3, right). The user can touch the map (*i.e.* create a new POI) or a circle (*i.e.* edit an existing POI) and fill its data through a dedicated form (Figure 4, middle). A color code allows distinguishing the user's records (*i.e.* red) from those of other participants (*i.e.* green). The users can interact with the POI data with its dedicated page (Figure 4, middle) *i.e.*: perform actions listed in section 3.3 and export their records to OSM. The user is informed of all rewards, in real-time thanks to a toast mechanism (Figure 3, right). All the rewards are gathered in the user profile page, accessible *via* the main menu (Figure 4, right): each mechanic has its own bar that can be touched to see its functioning and history.

The next section discusses the advantages and drawbacks of the current version of Albiziapp and our scenario model with an highlighting on the human learning aspect.

[12] https://www.nngroup.com/articles/ten-usability-heuristics/, *cf.* "consistency and standard" included in "Follow platform conventions"

5 Discussion and Conclusion

The Albiziapp architecture and descriptive model allow anyone creating a structured scenario, with highly customizable game mechanics for tree-based GCI. The system limitations lies in the JSON format requiring the understanding of its syntax. A more accessible HCI must be developed by, for example, the association of visual components for activities, actions and game mechanics [8]. Nonetheless, the choice of this format was related to a generator, under development, that can create JSON scenarios depending on the inventory, game and learning objectives. In addition, adding a new type of game mechanics is still a challenge with the current architecture.

Making relevant feedbacks regarding the user's progress in terms of tree identification skills is still an issue. We provide the [IDENTIFY] activity to set up some identification exercises (*cf.* section 3.2). Nevertheless, an expert is not necessarily available and this activity does not make a contribution to the inventory. Another strategy is to give to the user some points if another participant checks/edits her/his records and give similar information for the genus, the species or the common name. This has been implemented in our model *i.e.* in the JSON file, a score can be specified and linked to the respective action code: [sameGENUSPropagation], [sameSPECIESPropagation], [sameCOMMONPropagation] (*cf.* section 3.3). Obviously, this second option does not guarantee the veracity of the record data. However, it can be relevant in terms of "instant" feedback giving an insight of the user's identification skills. Nevertheless, the propagation formula could be enhanced as it does not take into account and detect, for example, the access to the record history by the participant making the verification.

This paper presents Albiziapp, a web application for the Geo-Collaborative Inventory (GCI) of trees. Keeping participants engaged in a GCI task is crucial to enhance the number of records, their completeness, their geo-distribution and veracity. Toward the variety of inventory situations, the questionable or non-existent feedbacks provided by the applications of the literature and the absence of scenarios guiding the user, a descriptive model of gamified scenarios was proposed. This model was built thanks to a simple structural analysis of the GCI task according to: (i) its goals and requirements and (ii) the macro-elementary user's actions. Each scenario, conformed to our model, can be formalized thanks to a JSON file that is operationalized by Albiziapp. To enhance the motivation and engagement of the participant, our descriptive model can specify four types of fully customizable reward-type mechanics at the activity level and/or the action level. In addition to the future work suggested in the previous section, a comparative study (GCI VS gamified GCI) will be conducted to analyse the influence of the gamification elements on: (a) the quantity and veracity of the recorded data during the inventory task and (b), the user's motivation, engagement and identification skills.

Acknowledgment: This work is supported by the French National Agency for Research with the reference ANR-15-CE38-0004-01 (ReVeRIES project).

References

1. Ahmad, A., Li, K., Feng, C., Asim, S. M., Yousif, A., Ge, S. . An empirical study of investigating mobile applications development challenges. IEEE Access, **6**, pp. 17711–17728, (2018)
2. Antoniou, V., Schlieder, C.: Addressing Uneven Participation Patterns in VGI Through Gamification Mechanisms. Geogames and Geoplay: Game-based Approaches to the Analysis of Geo-Information, pp. 91–110, (2018).
3. Cerutti, G., Tougne, L., Mille, J., Vacavant, A., Coquin, D.: Understanding Leaves in Natural Images - A Model-Based Approach for Tree Species Identification. Computer Vision and Image Understanding, **117**(10), pp. 1482–1501, (2013).
4. Dhuny, R., Sajeevan, G., Pardeshi, S.: An open framework to geotag and locate plants around the world. International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), pp. 153-158, (2015).
5. Filsecker, M., Hickey, D., T.: A multilevel analysis of the effects of external rewards on elementary students' motivation, engagement and learning in an educational game, Computers & Education, **75**, pp. 136–148, (2014).
6. Frisch, J. K., Unwin, M. M., Saunders, G. W.: Name That Plant! Overcoming Plant Blindness and Developing a Sense of Place Using Science and Environmental Education: The Inclusion of Environmental Education in Science Teacher Education, pp. 143–157, (2010).
7. Ahlqvist, O., Schlieder, C.: Geogames and Geoplay, Game-based Approaches to the Analysis of Geo-Information, Springer International Publishing, (2018)
8. Gicquel, P. Y., George, S., Laforcade, P., Marfisi-Schottman, I.: Design of a Component-Based Mobile Learning Game Authoring Tool. Games and Learning Alliance, pp. 208-217, (2017).
9. Han, J. Haihong, E. Le, G I& Du, J. Survey on NoSQL database. 6th IEEE international conference on pervasive computing and applications, pp. 363–366, (2011)
10. Nam, Y., Hwang, E., Kim, D.: CLOVER: A Mobile Content-Based Leaf Image Retrieval System. Digital Libraries: Implementing Strategies and Sharing Experiences, pp. 139–148, (2005).
11. Nguyen, T. T. N., Le, T. L., Vu, H., Hoang, V. S., Tran, T. H.: Crowdsourcing for botanical data collection towards to automatic plant identification: A review, Computers and Electronics in Agriculture", **155**, pp. 412–425, (2018).
12. Pimentel, V., & Nickerson, B. G.. Communicating and displaying real-time data with websocket. IEEE Internet Computing, **16**(4), pp. 45-53. (2012).
13. Roman, L. A., McPherson, E. G., Scharenbroch, B. C., Bartens, J.: Identifying Common Practices and Challenges for Local Urban Tree Monitoring Programs Across the United States. Arboriculture & Urban Forestry, **39**(6), pp. 292-299, (2013).
14. Skjoth, C. A., Geels, C., Hvidberg, M., Hertel, O., Brandt, J., Frohn, L. M., Hansen, K. M., Hedegaard, G. B., Christensen, J. H., Moseholm L.: An inventory of tree species in EuropeAn essential data input for air pollution modelling. Large Scale Computations in Environmental Modelling, **217**(3), pp. 292–304, (2008).