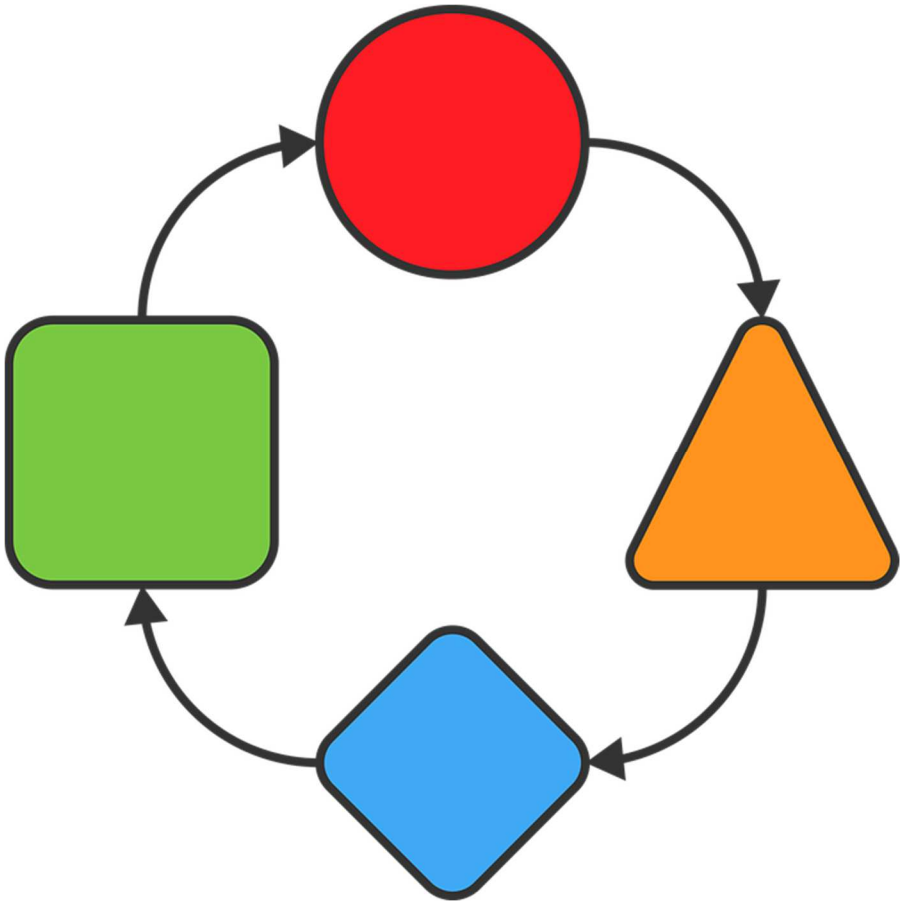
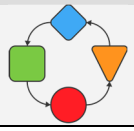
PROJECT VISION DOCUMENT



WQC



Version History

Data	Versão	Descrição	Autor	Revisor
25/10/19	1.0	Initial version	Richard Cardoso	-

Client Weger Walter GmbH

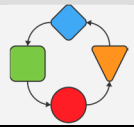
Document Project Vision Document: *WQC*

Date 25 October 2019

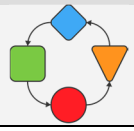
Author **Richard Cardoso**
richard.cardoso@weger.com.br

Signatures

Reviewed and Approved by: _____



Summary



1. Document goal

This document briefly describes the WQC system.

1.1. Scope

This document's scope is about the development of an application that suits the demands for quality check tasks informatization.

1.2. References

In order to prepare this document, the following references were used:

- Documents prepared by the author
  - Requirements document.pdf
  - Architecture diagram wqc.pdf
  - MarkSaveDiagram.pdf

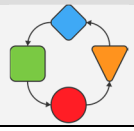
2. Business needs

An informatized application that allows to create quality reports based on existing documents as well as storing some pictures regarding the inspected objects to which the reports referees.

3. Project goal

Develop a web/mobile application capable of

- Store information at a MYSQL database
- Use HTTP protocol
- Run over any web browser (manager application)
- Run over Android 8.0 or newer (inspection application)



4. Preliminar Scope Statement

This section describes, at high level, the project scope. The requisites are going to be better explained at the requisites document.

4.1. Products to be delivered

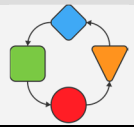
The following items are considered products of the project:

- Back-end code with a web interface that allow to:
  - Manage devices
  - Manage parameters of the system
  - Get a preview of the final reports
- Mobile app with an interface to work on the reports and take pictures

4.2. Requirements

4.2.1. Functional requirements

- The system should allow a qr-code to be scanned
- The system should retrieve from the server some pdf documents that belongs to the project to which the qr-code referees to
- The system should scan all the project's subfolders in order to retrieve all pdf documents that can be parsed to a report file (by having the filename beginning with one of the report types codes)
- The system should allow the user to add checks (marks) onto the reports
- The system should save a copy of the original pdf's and this copy should contain the marks added by the users
- The system should allow users to fill a predefined form of items with a state (approved, not approved, not applicable, not checked)
- For each of those items, the system should allow the user to take a picture which will refer to the corresponding item
- The system should store each one of those items state and properties
- The system should allow the user to inform data about the customer that requested this project
- The system should allow the user to take several project-related pictures (shared across all drawings and parts of the same project)
- The system should export/save all data (marked reports and taken pictures) to the fileserver
- The exported/saved data should always be up to date (it should be updated whenever a change is made by any user)
- The system should allow multiple users to work with the app at the same time (multi-user feature)
- The multi-user feature should work even when two or more users are working at the same entity (i.e. at the same report)
- The system should identify each one of the devices individually
- The system should allow devices to be registered / edited / enabled / disabled
- The system should allow to assign one or more roles to each device.
- Roles are defined as following:
  - All roles can capture general pictures.
  - TE: Technician. This role has read-only access to the project's contents.
  - MG: Magazine, EL: Electrician, QC: Quality-control: Those roles has read-write access to the project's contents.
- The assigned roles short label (MG, TE, …) should be shown up at each of the marks that the user adds to the reports surrounded.



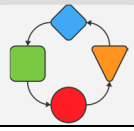
- Only authorized devices can access data from the projects
- All pictures should be saved inside the 'Fotos' folder of the corresponding project folder
- All reports generated by the system should be saved inside a folder named 'Qualitaetskontrol'. This folder should be located at the corresponding project folder (according to the qr-code)

4.2.2. Non-functional requirements

- The file upload feature should work at both windows/linux server
- The backend code must run at both windows/linux application server
- The mobile client must run under Android 8.0 or greater

4.2.3. Business rules

- Projects are composed of one or several devices/drawings
- Devices are composed of one or several parts
- Each device has one or several report files
- Each report can be either a report with predefined items or a pdf document
- Predefined items reports have a list of items whose status needs to be informed
- Predefined items report id is 5033
- Each item has: description, comments and a status
- Each item status can be either: Approved, Not approved or Not applicable
- Pdf document reports are represented by pdf files
- Pdf document report's files can be inside any subfolder of the project folder
- Each pdf report file starts with a 4-digits number that identifies the report type according to the following:
    - 5001: Non-wired drawing file
    - 5002: Non-wired datasheet file
    - 5032: Wired drawing file
    - 5033: Wired datasheet file
- Each part (physically speaking) is identified by a qr-code (attached to it)
- Every qr-code has the following structure: \xx-y-zzz_ Z_n_T_x, where:
    - xx: year in which the project has been executed
    - y: location (which company factory) in which the project has been executed
    - zzz: project number
    - n: drawing number
    - x: part number
- All project files are stored at a root folder named 'Auftrag', within a file server
- Inside the root folder, there is one folder for each past year and another for the current one
- Folders inside the root folder have a name that follows the following structure: xx-y-___, where:
    - xx: year in which the project has been executed
    - y: location (which company factory) in which the project has been executed
- Each of those folders contains a project folder whose name follows the following structure: xx-y-zzzz, where:
    - xx: year in which the project has been executed
    - y: location (which company factory) in which the project has been executed
    - zzz: project number
- Inside each project folder a lot of folders may exist. Among those there is a folder named 'Fotos', where the project's pictures should be stored.



5. Assumptions

- The system coding will be executed, primarily, at Weger's Brazilian site where:
  - Richard Cardoso (System's Developer):
    - Will be responsible to prepare the documentation, code and compiled/deployed final files to be used at production site
- The final products will be implemented at Weger's Headquarters in Italy where:
  - Kiener Paul (IT Admin):
    - Will provide relevant info (to the project scope) of the infrastructure
    - Will be responsible to do the implementation tasks upon receiving the updated files
  - Weger Thomas (Development engineer):
    - Will provide relevant info about the business rules
    - Will help to establish/track-down/polite the requisites for the system
    - Will carry out production tests

6. Architectural Representation

The systems will be developed based on the architecture illustrated at Image 1. All the architecture will be based on GoF and J2EE standards patterns. The back end will be executed inside an application server.

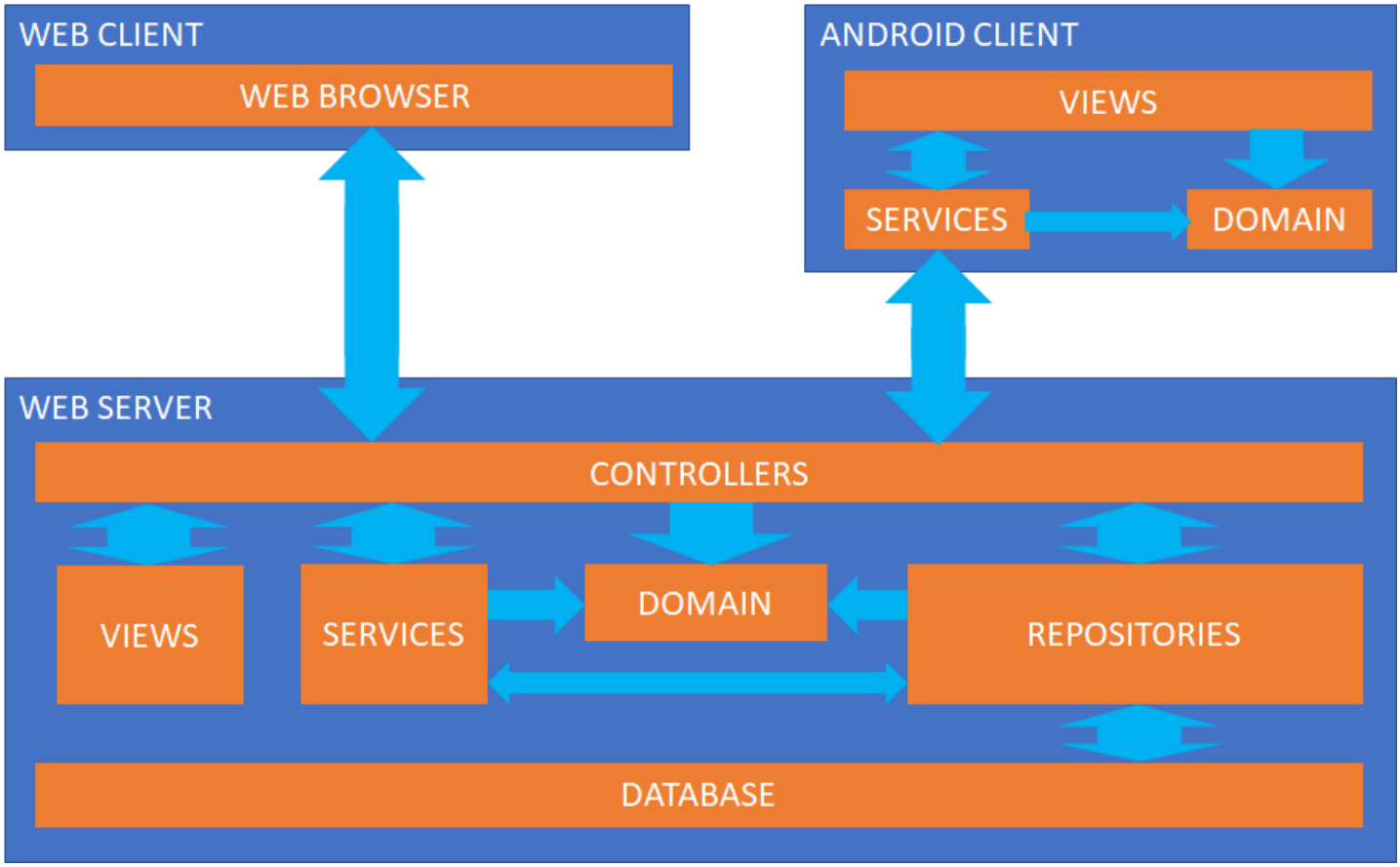


Image 1 - WQC Architectural Model

6.1. Web Server + Web client

a) The views are responsible for the presentation layer. The Technologies used in this layer will be:

- HTML
- Thymeleaf
- Bootstrap

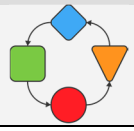
b) The controllers will be responsible for receiving the client's data, deliver it to the appropriated services, get the services response, attach the response to the appropriated view and then deliver the view with the response to the clients. The technologies used in this layer will be:

- Spring MVC

c) The services are responsible for processing the data received from the controllers and deliver the appropriated responses. The technologies used in this layer will be:

- Spring MVC

d) The domain is a set of POJO objects that represents the value objects of the system.



e) The repositories are responsible for abstracting the database procedures and operations. The technologies used in this layer will be:
   - Spring Data JPA

f) The database is used by the system to store/retrieve data. The technologies used in this layer will be:
   - MySQL

6.2. Android client

a. The views are responsible for the presentation layer. This layer will be implemented using android's activities

b. The services are responsible for processing the data received from the views and deliver to them the appropriated responses. This does includes acting as a middleware between the views and external services.

For the whole system GoF patterns will be used whenever it is appropriated. Some of them are: Faccade, Observer and Singleton.

6.3. Architectural restrictions

There were identified some restrictions that are pertinent to the development of this subsystem:

- Utilization of Java's JDK 8
- Utilization of Apache Tomcat 8.5

7. Use-Case vision

This section presents the general use cases of WQC

7.1. General use-case diagram for the Android clients

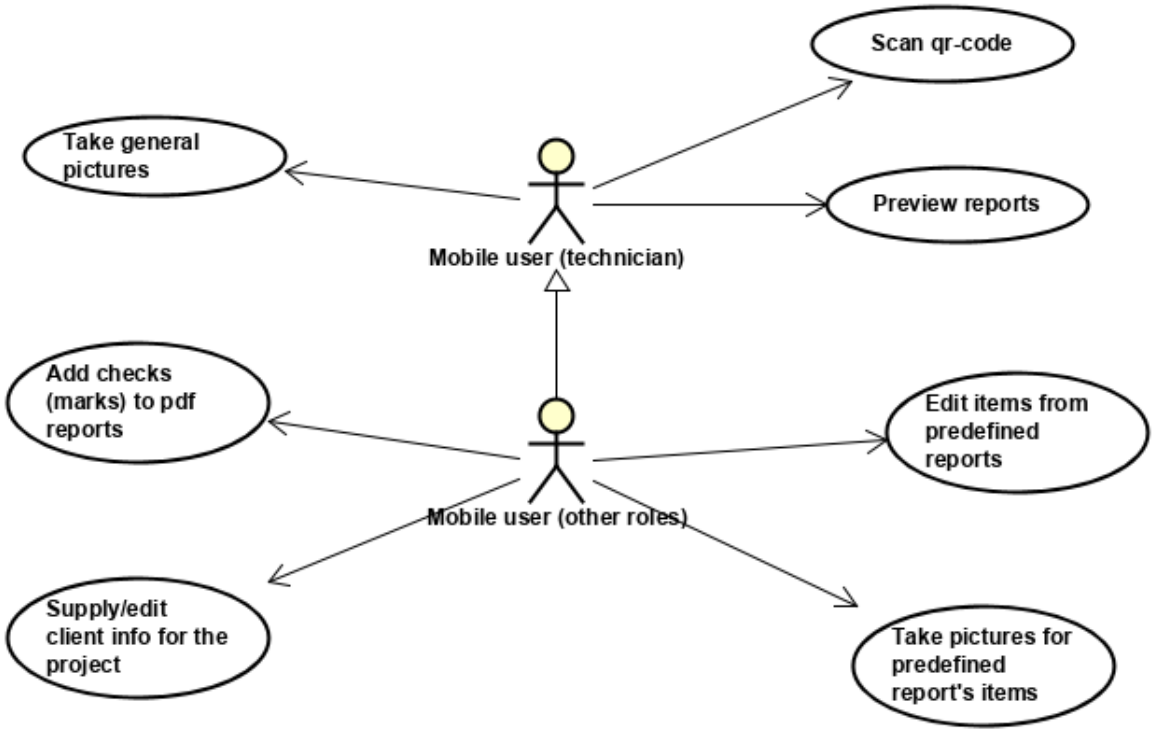


Image 2 – General use-case diagram for the android clients

7.2. General use-case diagram for the web clients

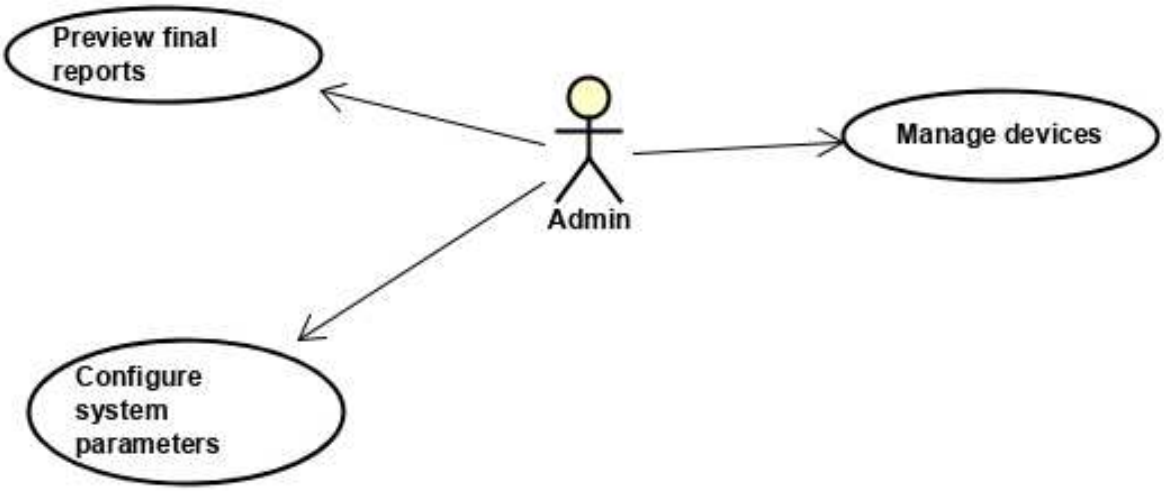
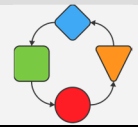


Image 3 – General use-case diagram for the web clients



8. Domain vision

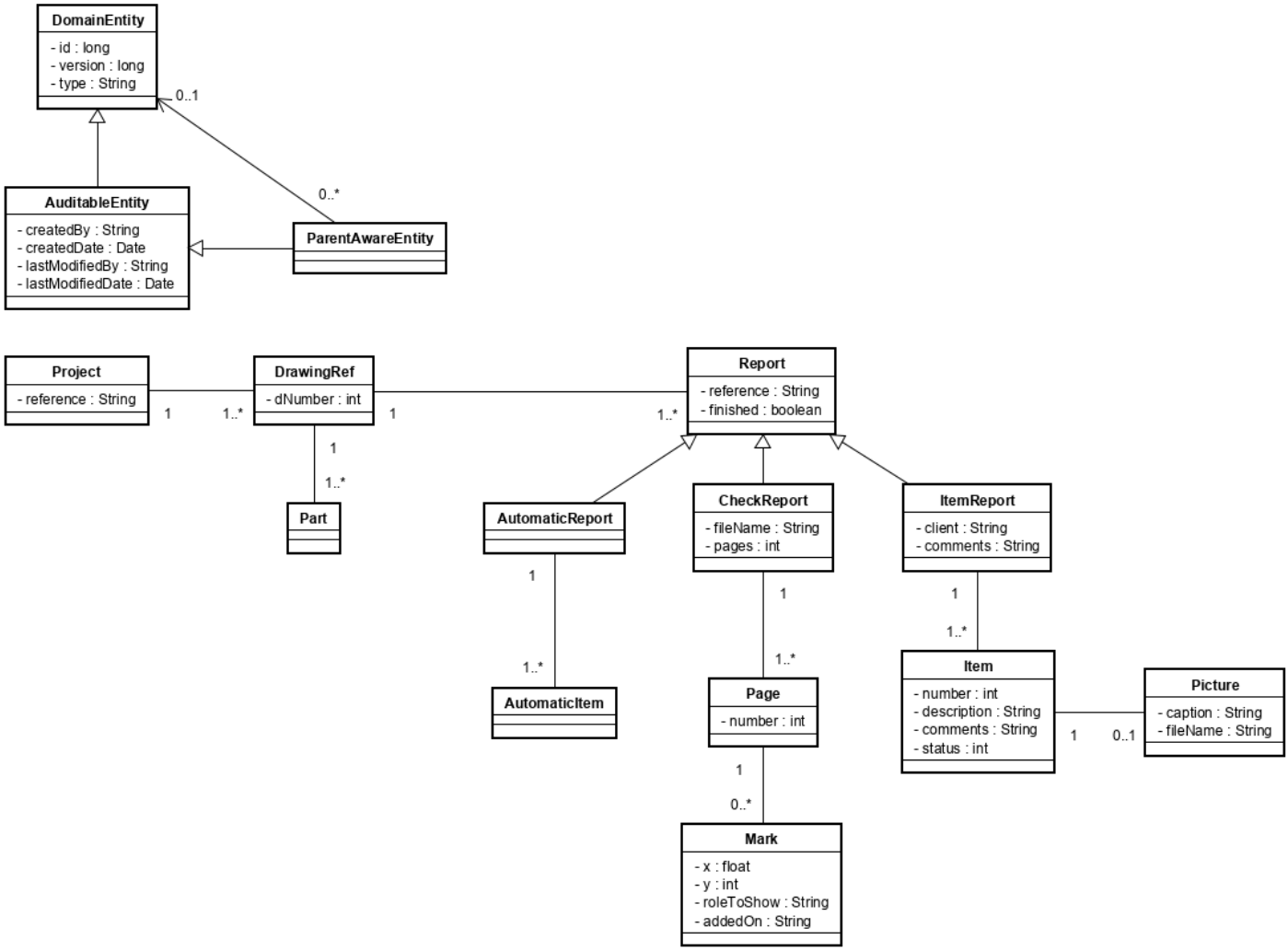


Image 4 – Main domain classes

9. Use-case realizations

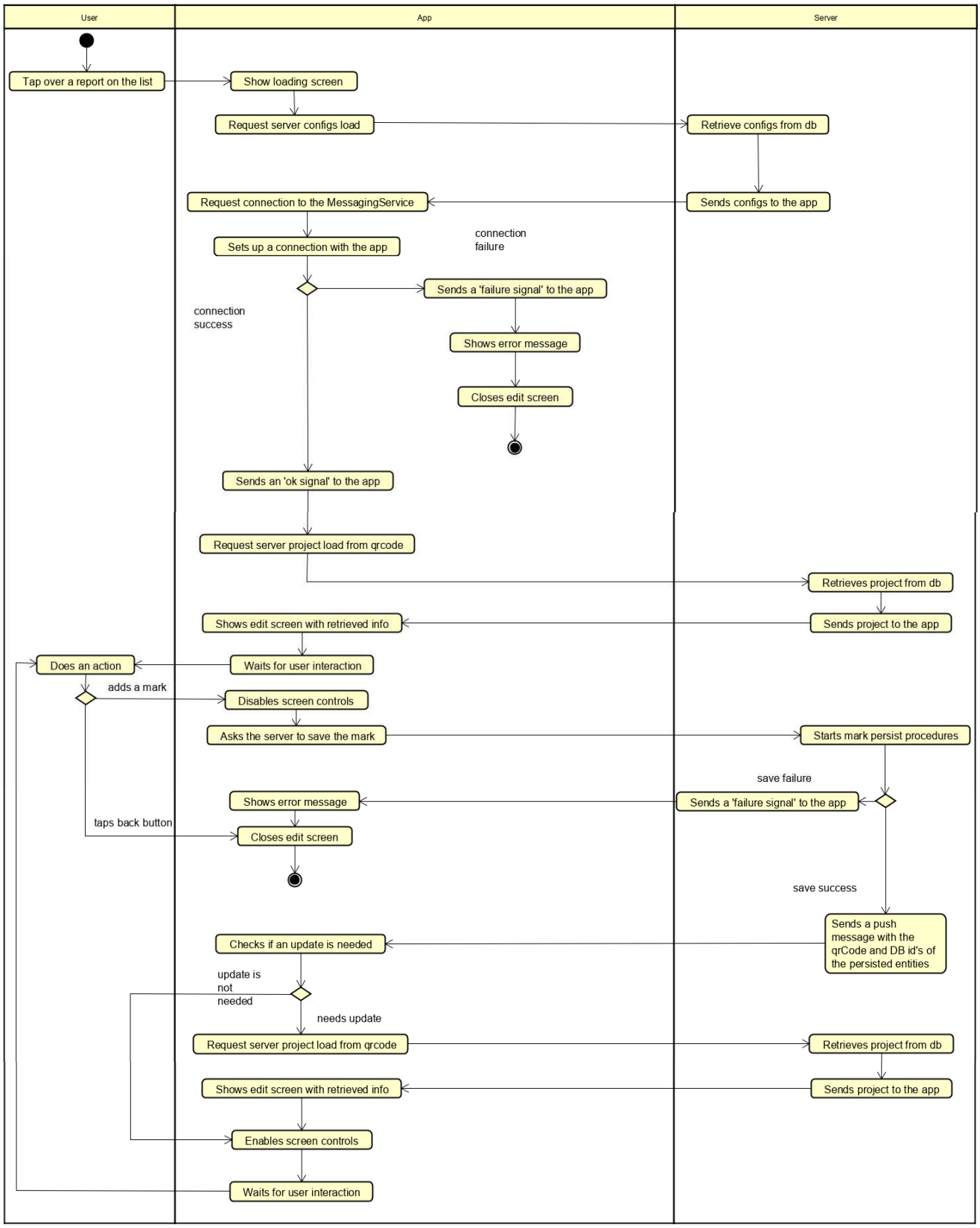
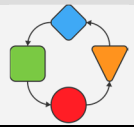


Image 5 – Activities-diagram for the mark saving process



10. References

Unified Modeling Language: http://www.omg.org/technology/documents/formal/uml.htm