



TODO

Engenharia Informática e Computadores
Projecto e Seminário - 2020/21

G35

Bernardo Rodrigues, n. 42130
bernardo.qtr.21@gmail.com

Supervisors:
Nuno Leite, ISEL, nuno.leite@isel.pt

1 Introduction

Today's world is a fast, complex moving mess. As such we try to schedule our days or weeks, or we create small reminders to do something, whatever that may be. Those little notes we give ourselves, those "to-do's" are often forgotten and not fully fulfilled. As such this application will attempt to mend that by creating an environment where a reminder is easily created but hardly forgotten or skipped over.

2 Proposed Solution

The goal of the project is to create a web application using current practices and technologies. The application will be comprised of a web *API*, a database and front-end application as shown below. The proposed solution is intended to be generic and expandable for fu-

Fig. 1: Abstract architecture of the solution

ture use. The user will be able to create, delete and edit new reminders or "to-do's" and assign them a priority as well as a date. The application will then rank these to-do's and arrange them in a way where the user will not forget them and will actually fulfill them. The web *API* will be a *REST API* composed of 4 layers. The controllers (endpoints), the business layer, the third-party *API* consumers (ie the solvers) and the data access layer (*DAL*) as is shown in the image below. This will be done using the *.NET* [?] and *MVC* [?] frameworks. The *DAL* will be using the framework Hibernate [?] as a way to simplify and improve future development. The database will be constructed using *SQL Server*[?]. The front-end development will be using *Angular 9*[?].

3 Required Functionalities

1. Allow users to sign up, log in, log out and delete the account.
2. Create, edit and erase to-do's.
3. Push notifications.
4. Rank to-do's based on date and priority.
5. Image recognition creation based on physical to-do's (English only).

4 Optional Functionalities

1. Different to-do lists based on categories.
2. *OpenId* [1] log in.
3. Profile Picture.

5 Schedule

References

- [1] Brad Fitzpatrick. OpenId Connect, 2007. <https://openid.net/>.