

Architecture Notebook

Project

UARobotFight

Team

UARobotFight

Author(s)

name	id	email
Miguel Azevedo	38569	lobaoazevedo@ua.pt

Revision(s)

< contains the list of changes - when, who and what was changed >

date	Who	What
March 5, 2013	Miguel Azevedo	First Version
March 19, 2013	Miguel Azevedo	Second Iteration

Table of contents

- [1. Purpose](#)
- [2. Architectural goals and philosophy](#)
- [3. Assumptions and dependencies](#)
- [4. Architecturally significant requirements](#)
- [5. Decisions, constraints, and justifications](#)

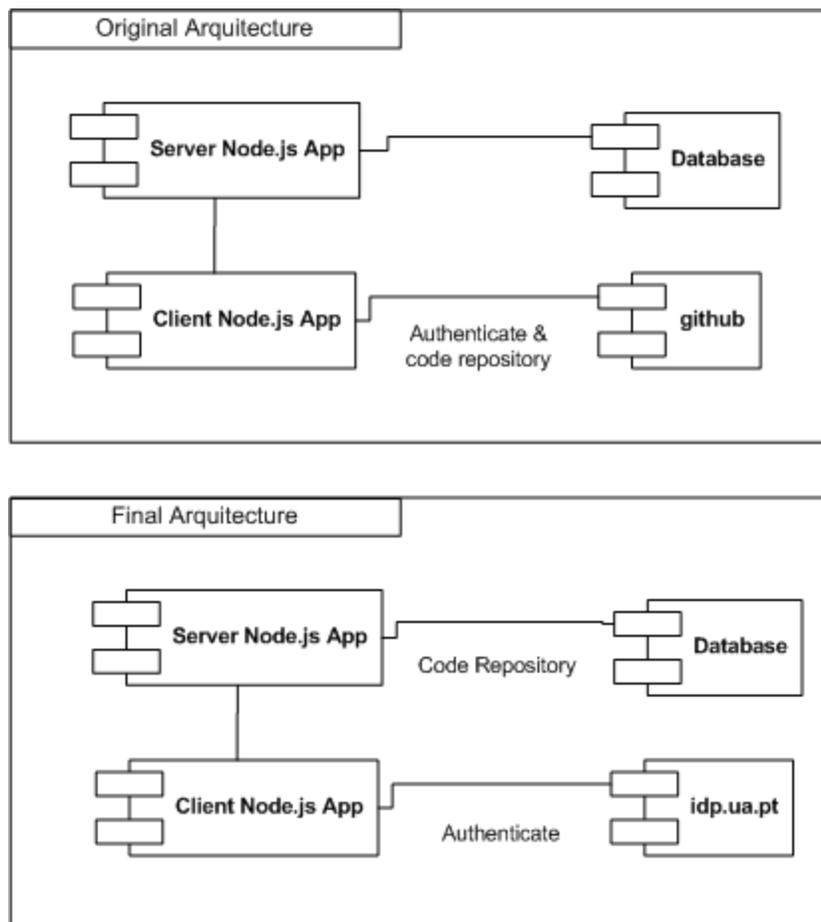
1. Purpose

This document describes the philosophy, decisions, constraints, justifications, significant elements, and any other overarching aspects of the system that shape the design and implementation.

2. Architectural goals and philosophy

The first implementation (the original FightCodeGame) consists of a Node.js server application and a Node.js client application which communicate with github for authentication and code repository.

The final solution is supposed to be a Node.js application (a fork of the original FightCodeGame) featuring idp.ua.pt based authentication and using its “internal” PostgreSQL database.



3. Assumptions and dependencies

The afore mentioned idp.ua.pt authentication feature will require an integration of the application (UARobotFight) accordingly with idp.ua.pt's guidelines, since the initial solution uses a PostgreSQL database for application data storage which already features robot code repository facilities, this database will be used virtually same way FightCodeGame uses github, as a code repository for users' robots.

4. Architecturally significant requirements

As previously stated, an interface with idp.ua.pt authentication service will have to be implemented. An interface with the database for its use as a code repository will have to be implemented.

5. Decisions, constraints, and justifications

- The final solution should feature an idp.ua.pt based authentication, since its purpose is to be used by the university's students.
- The final solution will not use code.ua.pt the same way the original solution uses github since code.ua.pt doesn't provide the same interaction facilities and API's which github does. Alternatively the server application's internal PostgreSQL will be used as well as a code repository for the users' robots.
- Several other feature additions may impact the architecture at a smaller scale which as an example may imply the existence of a new module for battle logging(battle replay

feature) or imply the existence/modification of new data-structures for representing weapons, maps, etc.