

**Curso de Engenharia Informática**

**Engenharia de Software**

**Relatório do Projeto Semestral**

**Alessandro Miguez 27583**

**João Azevedo 26394**

**20 de Janeiro de 2015, Universidade Fernando Pessoa, Porto**

Relatório

# Requisitos Funcionais

Foi pretendido fazer dois web services:

1. Deveria permitir
   1. Registo de utilizadores;
   2. Registo de atividades;
   3. Consulta das mesmas.
2. O segundo web service deveria permitir
   1. Inserção de dois tipos de utilizadores (Pacient e Care Taker);
   2. Guardar referência onde a informação de um Pacient está guardada;
   3. No papel de Pacient autorizar um dado Care Taker a visualizar determinada informação;
   4. Um Care Taker tem acesso a uma dada informação;
   5. Registar histórico de atividades;
   6. Criar regras de anomalias baseado no histórico de atividades;
   7. Alertas automatizados de anomalias.

# Obstáculos

1. Adaptação às novas ferramentas (nusoap, git, doxygen, project);
2. Adaptação às linguagens de desenvolvimento (php, mysql, python);
3. Adaptação às técnicas de desenvolvimento de software (adequação à metodologia prototipagem e às notações dos diversos diagramas necessários)

# Tarefas Realizadas

Do primeiro ws, conseguimos fazer:

1. Inserção de utilizadores;
2. Inserção de atividades;
3. Consulta de atividades

Do segundo ws:

1. Inserção dos tipos de utilizadores;
2. Inserção das referências das informações dos pacientes;
3. Atribuição de autorização de uma atividade a um Care Taker;
4. Acesso de um Care Taker a uma dada informação.

# O que se pode melhorar

No primeiro ws:

1. Melhorar a experiência de utilizador;
2. Implementar sistema de filtragem na pesquisa;
3. Aumentar a diversidade de informações;
4. Verificação dos dados inseridos pelo utilizador no lado servidor.

No segundo ws:

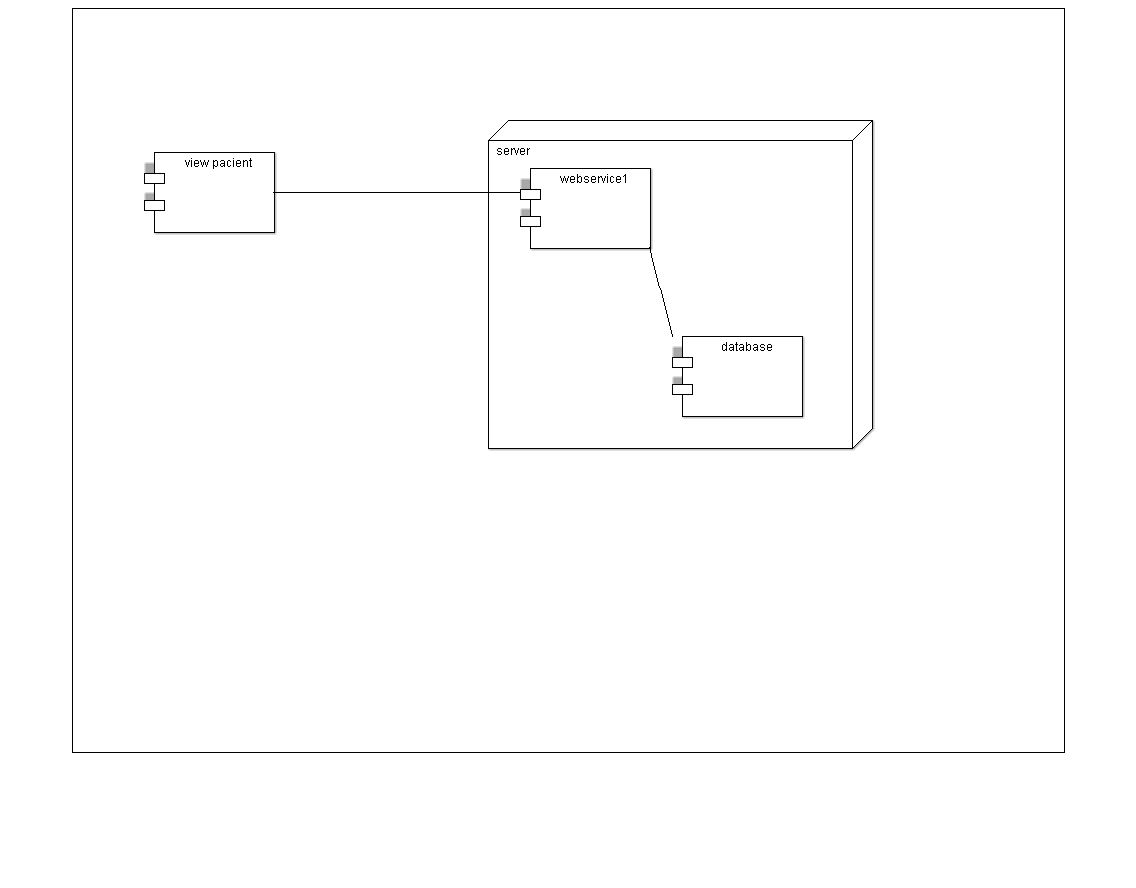
1. Implementar os requisitos não concluídos;
2. Criação de lista de WS suportados pela plataforma;
   1. Listagem dinâmica desses mesmos web services quando paciente regista uma atividade.
3. Melhora a experiência de utilizador;
4. Verificação dos dados inseridos pelo utilizador no lado servidor.

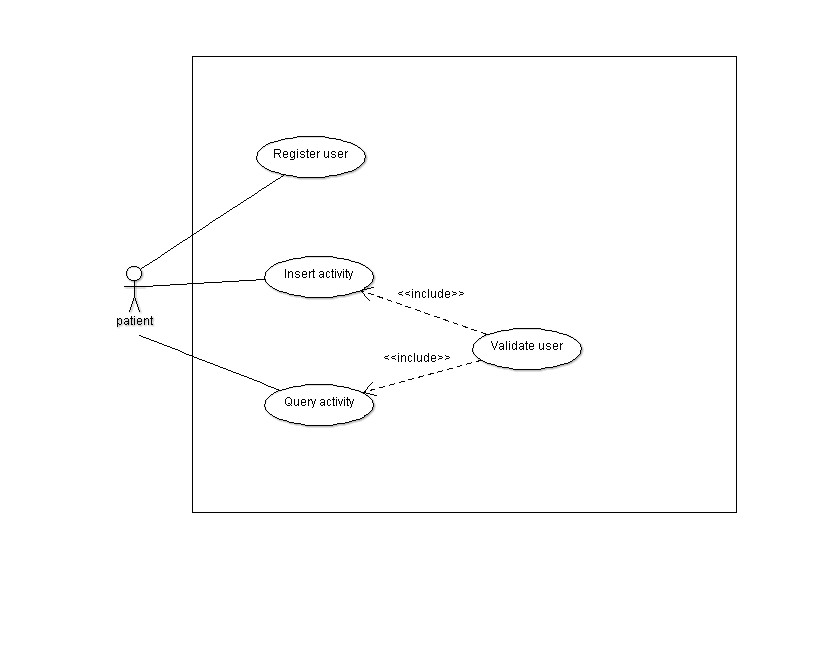
# Recursos Utilizados

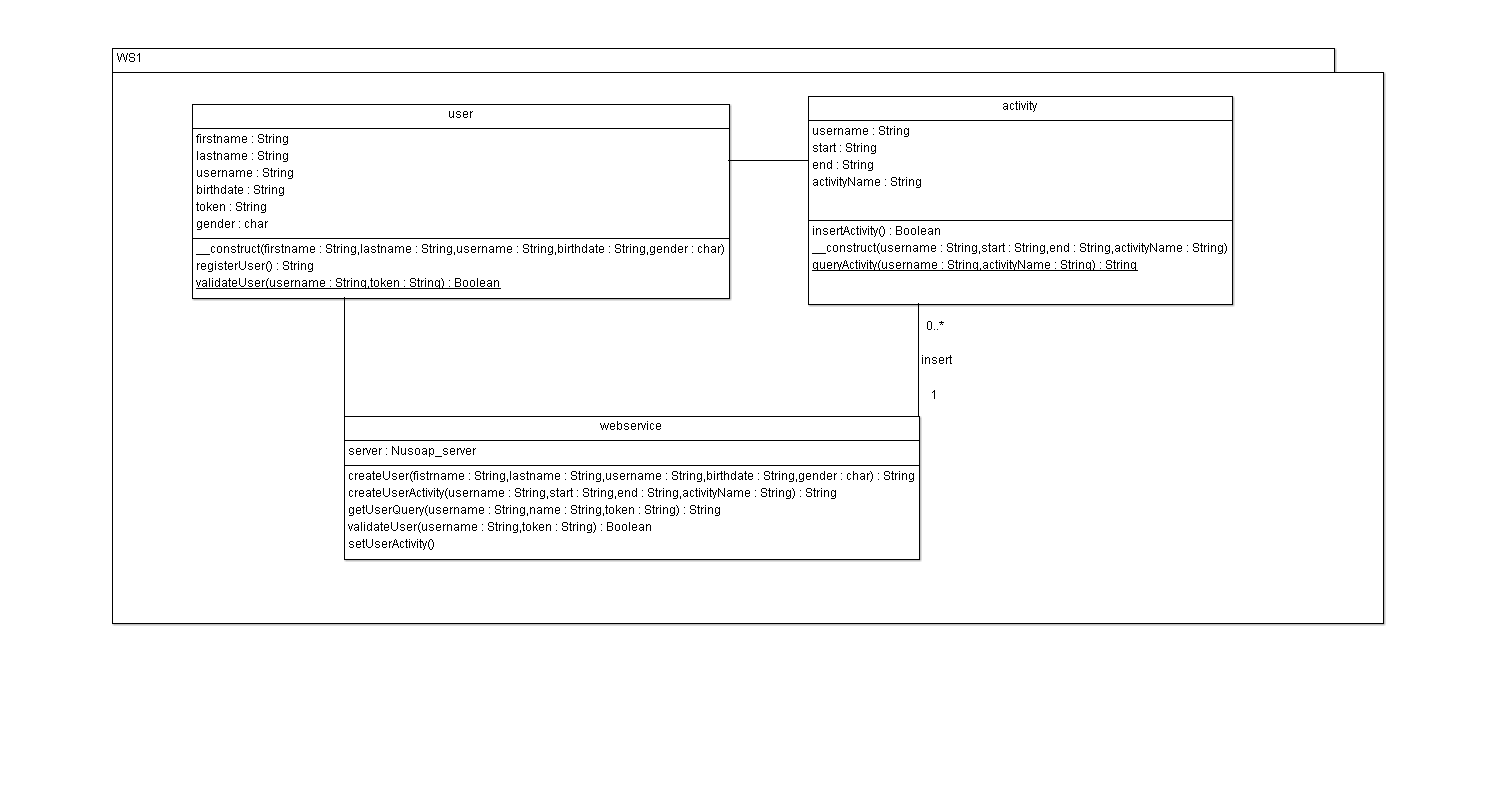
1. Office Project;
2. Bitbucket (https://bitbucket.org/alessandro\_miguez/es-project);
3. Argo UML;
4. Editor de texto;
5. Emulador de servidor.

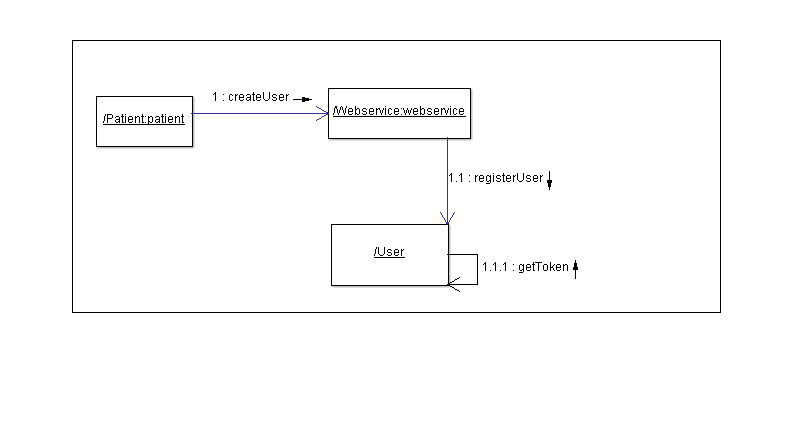
# Diagramas

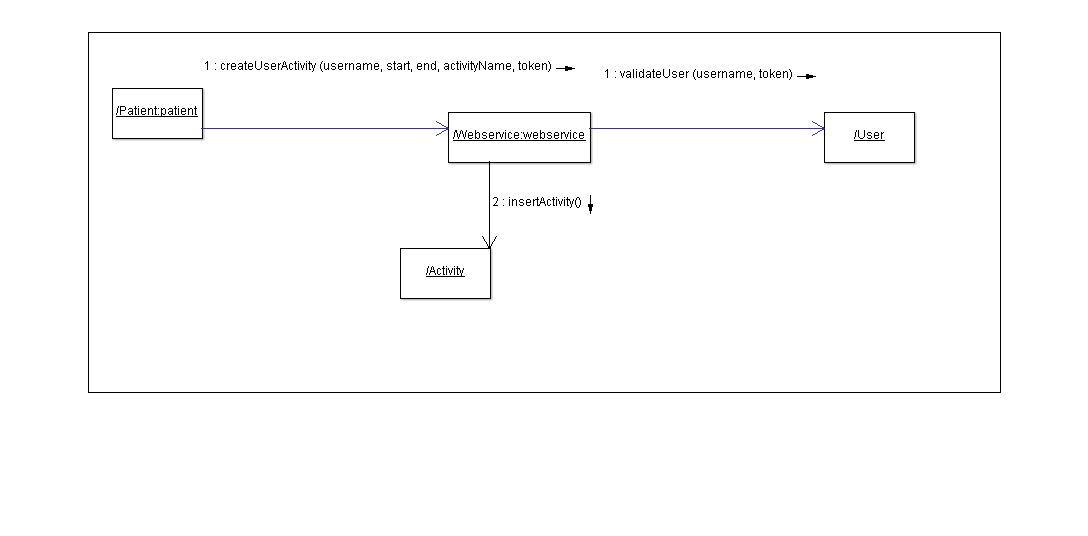
1. Diagramas relativos ao Webservice 1

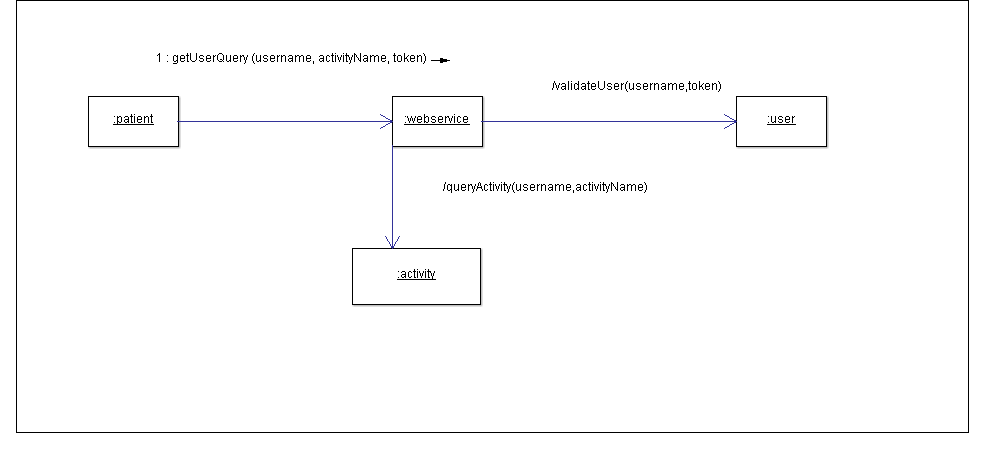












1. Diagramas relativos ao webservice 2

