

SISTEMA DE GESTÃO DE PROJETOS DE SOFTWARE - SGPS

**Lilian R. M. Paiva , Luciene C. Oliveira, Mariana D. Justino, Mateus S. Silva,
Mylene L. Rodrigues**

Engenharia de Computação - Universidade de Uberaba (UNIUBE)

Av. Rondon Pacheco 2000, Bairro Lídice – Uberlândia – MG– Brasil

{lilian.paiva, luciene.oliveira, mariana.davi, mateus.silvestre,
mylene.rodrigues}@ sgps.eng.br

***Abstract.** Software Engineering is a field of computing knowledge very important for the software production, which includes since specification, development, testing until maintenance of systems by applying technologies and practices of project management and other disciplines, which aims to organize, productivity and quality. The process of building software requires constant project management software. As a result, easier becomes the need for tools to make the project management software. The objective of this work is to propose a model of Project Management System Software (SGPS) which aims at achieving control of the development of software.*

***Resumo.** A Engenharia de Software é uma área do conhecimento da computação muito importante para a produção de software, que contempla desde a especificação, desenvolvimento, testes até a manutenção de sistemas aplicando tecnologias e práticas de gerência de projetos e outras disciplinas, o que objetiva na organização, produtividade e qualidade. O processo de construção de software necessita de um constante gerenciamento de projeto de software. Em virtude disso, surge a necessidade de ferramentas para facilitar a gestão de projeto de software. O objetivo deste trabalho é propor um modelo de Sistema de Gestão de Projetos de Software (SGPS) que visa realizar o controle das etapas do desenvolvimento de software.*

1. Introdução

A Engenharia de Software é uma área de extrema importância no mundo da computação já que é uma disciplina de engenharia relacionada com todos os aspectos da produção de software, desde os estágios iniciais de especificação do sistema até sua manutenção, depois que este entrar em operação [Sommerville 2006].

Gestão de Projetos, uma área da engenharia de software, é um termo que cresce continuamente nas empresas nos dias atuais que consiste em metodologias, técnicas, conhecimentos técnicos, habilidades e ferramentas na condução das atividades da produção de um projeto de software visando atender os seus objetivos. [PMI 2004]

Diante das necessidades na área de Engenharia de Software e Gestão de Projetos, o desenvolvimento de um software para realizar esta gestão faz-se necessário. Portanto, o objetivo deste artigo é propor um modelo de um sistema de Gestão de Projetos de Software

(SGPS) para auxiliar no controle de todas as fases do desenvolvimento de software: Especificação de Requisitos, Projeto, Desenvolvimento, Testes e Implantação.

2. Conceitos Básicos

Neste item serão apresentados os principais conceitos e tecnologias utilizadas para o desenvolvimento deste projeto.

2.1. Gestão de Projetos

De acordo com [Sotille, 2004], o gerenciamento de projetos de software é uma parte essencial da engenharia de software, composto por algumas das seguintes áreas de gestão:

1. **Gestão de Tempo:** é necessário elaborar um cronograma, identificando os marcos, atividades e recursos para o projeto;
2. **Gestão de Custos:** o gerenciamento de orçamento é necessário para atender as metas da empresa que está desenvolvendo o software;
3. **Gestão de Atividades:** é muito importante gerenciar as atividades realizadas pelos recursos envolvidos em um projeto de software, bem como a produtividade das pessoas;
4. **Gestão de Mudanças:** as mudanças são inevitáveis. Assim que o software é colocado em uso, novos requisitos surgem e os requisitos existentes são alterados à medida que a empresa passa por modificações. Devido às mudanças ocorridas, é importante realizar o gerenciamento de mudanças;
5. **Gestão de Riscos:** área que identifica, qualifica, controla e quantifica os riscos do projeto. O gerenciamento de riscos está sendo considerado, cada vez mais, como uma das principais atividades dos gerentes de projeto;
6. **Gestão de Implantação:** processo responsável pelo planejamento, programação, armazenamento e liberação de todo software e hardware autorizado para os ambientes de testes e produção dentro de uma organização.

A gerência de projetos é a primeira camada do processo de engenharia de software. Ela abrange todo o processo de desenvolvimento de software, do começo ao fim.

2.2. A Linguagem de Modelagem UML

A UML (Unified Modeling Language) é uma linguagem visual utilizada para modelar softwares baseados no paradigma de orientação a objetos. É uma linguagem de propósito geral que se pode ser aplicada a todos os domínios de aplicação [Guedes 2009].

Nas fases de concepção e elaboração do sistema foi elaborada uma documentação de requisitos e arquitetura contendo modelos de software utilizando a linguagem UML.

2.3. A Tecnologia Java

A linguagem Java é uma poderosa linguagem de programação orientada a objetos para desenvolvimento de aplicações *Web*, *Desktop* e para dispositivos móveis.

Para a fase de construção do sistema são utilizados os conceitos do paradigma de desenvolvimento orientado a objetos, bem como, desenvolvimento web utilizando a linguagem Java, plataforma JEE (*Java Enterprise Edition*), Banco de Dados e *frameworks* de desenvolvimento, JSF (*Java Server Faces*), *Hibernate*, *Spring*, entre outros.

3. SGPS – Sistema de Gestão de Projetos de Software

O sistema de gestão de projetos de software (SGPS) está dividido em módulos do gerenciamento de projetos e da engenharia de software, descritos a seguir.

3.1. Gestão de Requisitos

Os requisitos de um sistema são descrições dos serviços fornecidos pelo sistema e as suas restrições operacionais. O processo de analisar, validar, verificar, documentar e restringir é chamado de Engenharia de Requisitos [Sommerville 2006]. O módulo de requisitos compreende em contemplar no sistema toda a especificação de requisitos de um software com o objetivo de estabelecer uma visão comum entre o cliente e a equipe do projeto em relação aos requisitos que serão atendidos pelo software. Na Figura 1, é apresentado o diagrama de caso de uso contendo as funcionalidades do módulo de Gestão de Requisitos.

De acordo com o caso de uso, o analista de requisitos identifica os requisitos do projeto e cadastra no SGPS, integrando com as informações já cadastradas, definidas pelo gerente de projeto e diretor da empresa.

3.2. Gestão de Tempo

O módulo de gestão de tempo compreende na elaboração do cronograma, contendo as atividades e recursos envolvidos em um projeto de software.

A gestão de atividades é acoplada ao módulo de gestão de tempo, pois contempla gerenciar as atividades realizadas pelos recursos envolvidos em um projeto de software. Neste módulo, um conjunto de tarefas é identificado e interdependências entre as tarefas são estabelecidas, conforme pode ser visualizado no diagrama de caso de uso apresentado na Figura 2.

O módulo de gestão de custos está contido no módulo de gestão de tempo, pois, de acordo com [Pressman 2004], a estimativa dos recursos, custos e a programação das atividades para um esforço de desenvolvimento de software exige experiência e acesso a boas informações históricas.

De acordo com o caso de uso, o gerente do projeto irá gerenciar o cronograma do projeto, bem como os custos e recursos disponíveis.

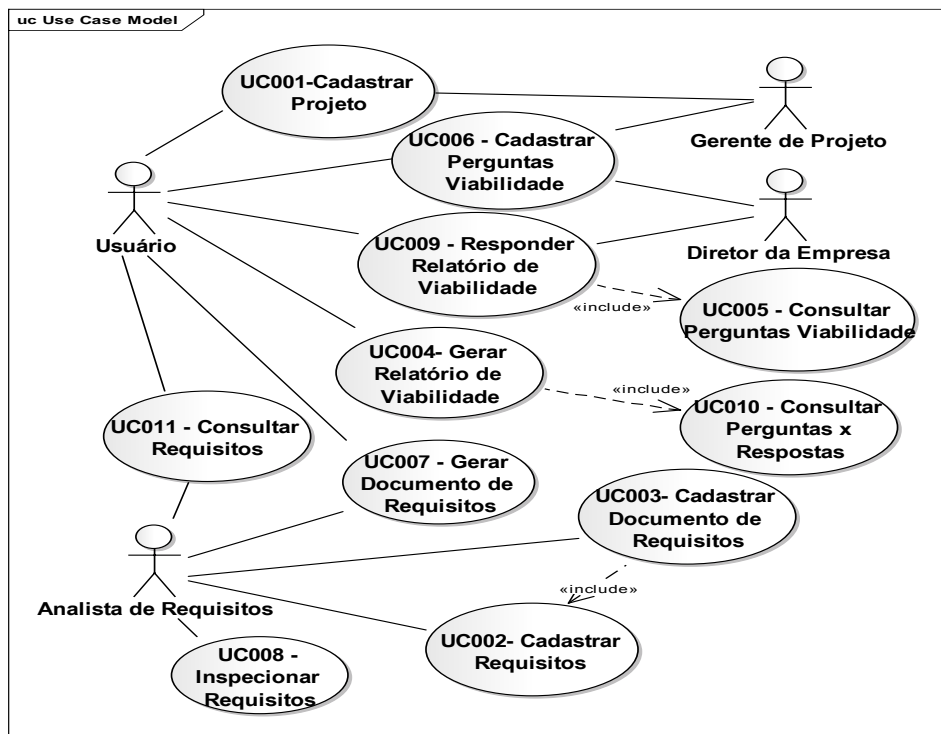


Figura 1. Casos de uso do módulo de Gestão de Requisitos

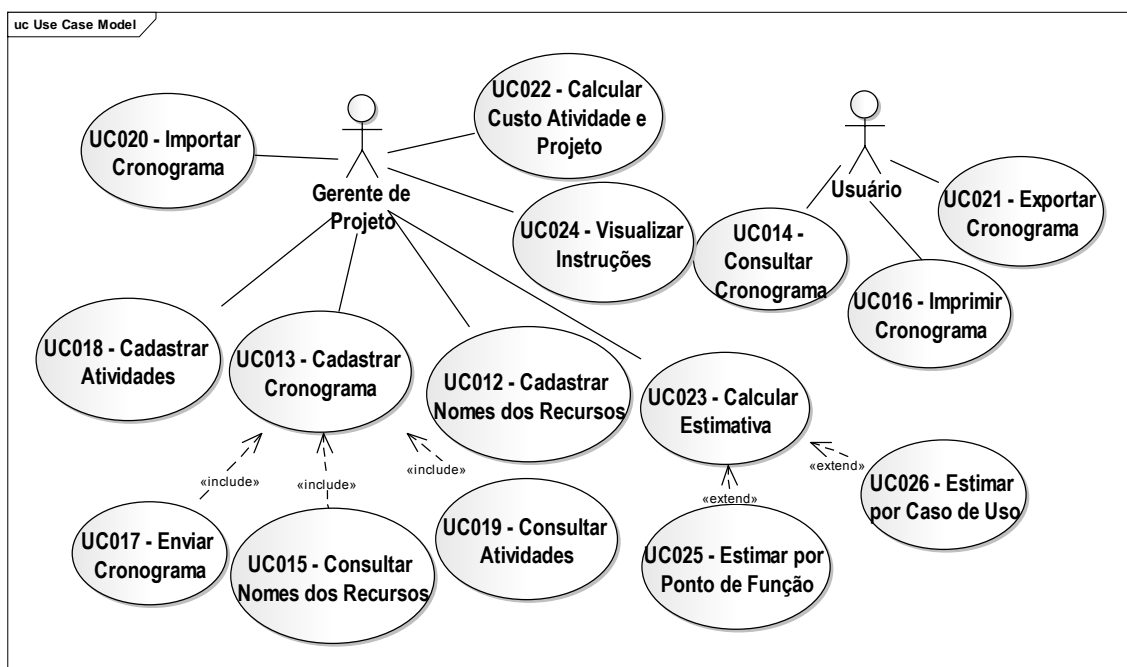


Figura 2. Casos de uso do módulo de Gestão de Tempo

3.3. Gestão da Arquitetura

O módulo de gestão de arquitetura de projeto de software (Figura 3) compreende a inclusão de diagramas, modelos e artefatos pertencentes à fase de projeto de um software. Neste módulo, os requisitos são traduzidos em uma representação de software.

Este módulo representa uma etapa crucial para o sucesso no desenvolvimento de qualquer sistema, pois com ele o projetista tem uma visão ampla do que deve ser feito e aplica a estratégia que melhor atende às necessidades do software.

De acordo com o caso de uso do módulo de gestão de arquitetura, o arquiteto definirá juntamente com o cliente, o desenvolvedor e o gerente a arquitetura a ser usada no projeto em questão, relacionando com os dados cadastrados previamente pelo administrador do sistema com o usuário.

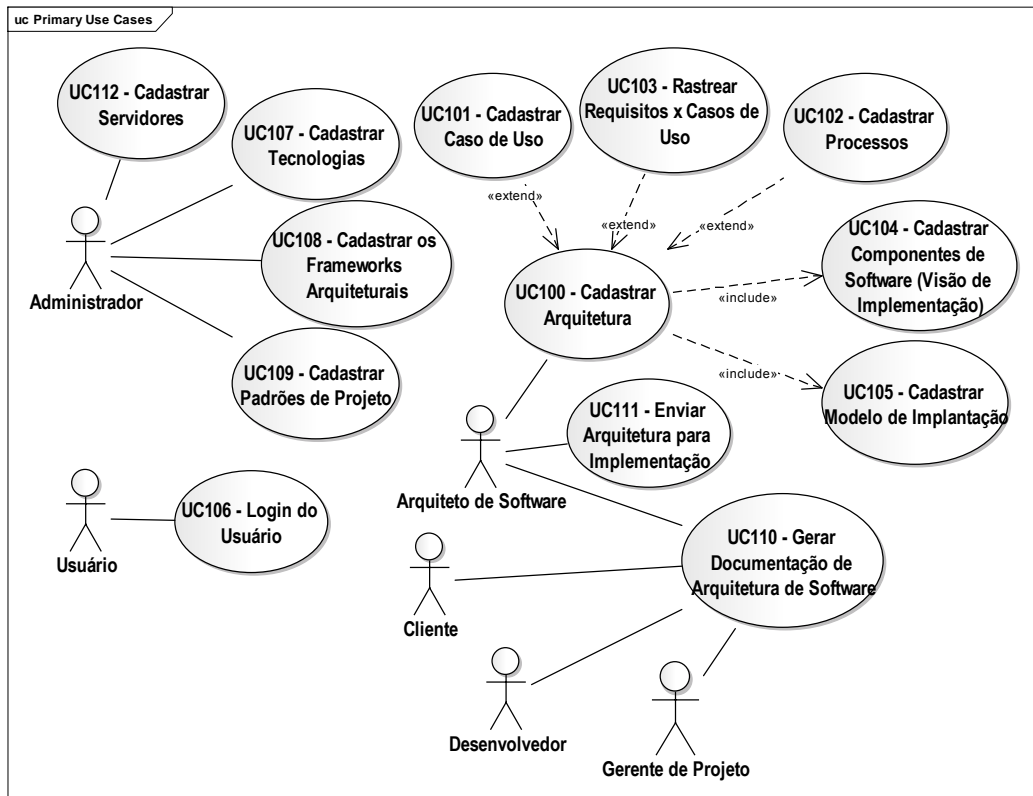


Figura 3. Casos de uso do módulo de Gestão da Arquitetura

3.4. Gestão de Mudanças

Outro módulo do SGPS é o de gestão de mudanças, que compreende o gerenciamento das mudanças ocorridas durante e após o desenvolvimento de um software. É nesta fase que se decide se é melhor ou não aplicar a mudança solicitada ao projeto. O caso de uso pode ser verificado na Figura 4.

Conforme caso de uso de gestão de mudanças, o usuário em conjunto com o cliente, desenvolvedor e gerente de mudanças, definirá e aprovará as alterações necessárias, cadastrando-as no módulo em questão.

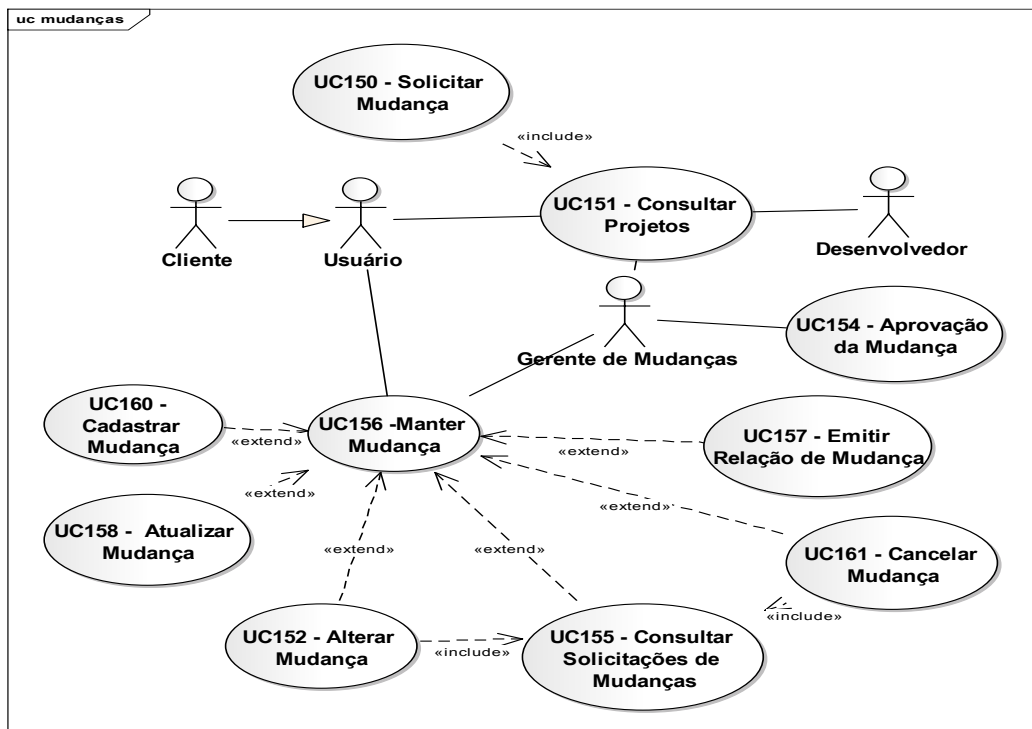


Figura 4. Casos de uso do módulo de Gestão de Mudanças

3.5. Gestão de Testes

Segundo Pressman [Pressman 2004], a qualidade de software é responsável por definir um conjunto de atividades que irão ajudar a garantir que cada produto de trabalho da Engenharia de Software exiba alta qualidade. O módulo de gestão de testes contempla todas as etapas e fases do processo de testes, por exemplo, teste de unidade, teste de aceitação e teste de integração e é descrito através do caso de uso da Figura 5.

Conforme caso de uso da gestão de testes, o líder técnico é o responsável por executar os testes unitários, de código e integrar com ferramentas de testes, de acordo com a necessidade. O analista de testes cadastra os casos a serem testados, para que o testador execute e o cliente dê o aceite final.

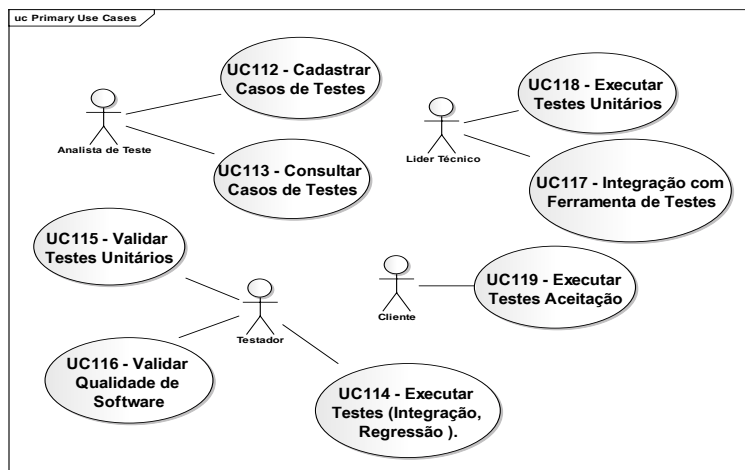


Figura 5. Casos de uso do módulo de Gestão de Testes

3.5. Gestão de Riscos

O módulo de gestão de riscos compreende a análise dos riscos do projeto. Essa análise é crucial para uma boa gestão de projetos e envolve sua identificação, avaliação, disposição por ordem de prioridade, estratégias de administração, resolução e monitoração.

Conforme caso de uso da gestão de riscos, o usuário, juntamente com o administrador do sistema monitora os riscos além de planejar ações corretivas e preventivas.

As funcionalidades desse módulo são descritas através do caso de uso da Figura 6.

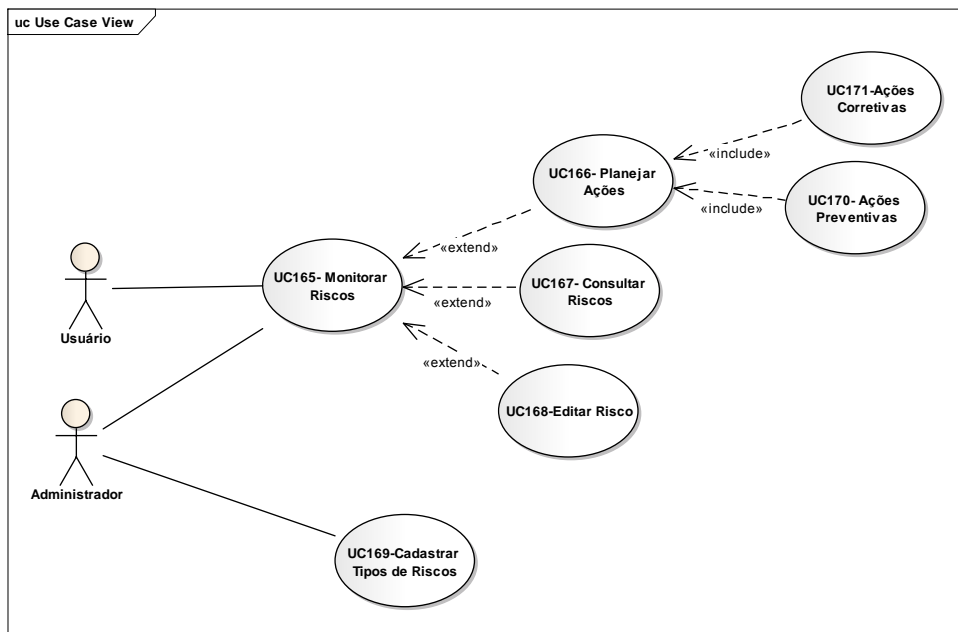


Figura 6. Casos de uso do módulo de Gestão de Riscos

3.6. Gestão de Implantação

O gerenciamento de implantação ou gerenciamento de liberação é o processo responsável pelo planejamento, programação e controle das liberações do software para os ambientes de testes e produção. O módulo de implantação compreende em auxiliar na implantação de um projeto de software. Este módulo representa todos os processos necessários para colocar o sistema em operação e também contempla o gerenciamento dos objetos e artefatos que foram construídos no sistema.

De acordo com o caso de uso da gestão de implantação, o desenvolvedor junto com o suporte do cliente controla as liberações do software para os ambientes de testes e produção, armazena as informações dos artefatos e objetos construídos no projeto de software, bem como o gerenciamento de configuração contendo o armazenamento dos *builds* e versões geradas do software.

As funcionalidades desse módulo são descritas através do caso de uso da Figura 7.

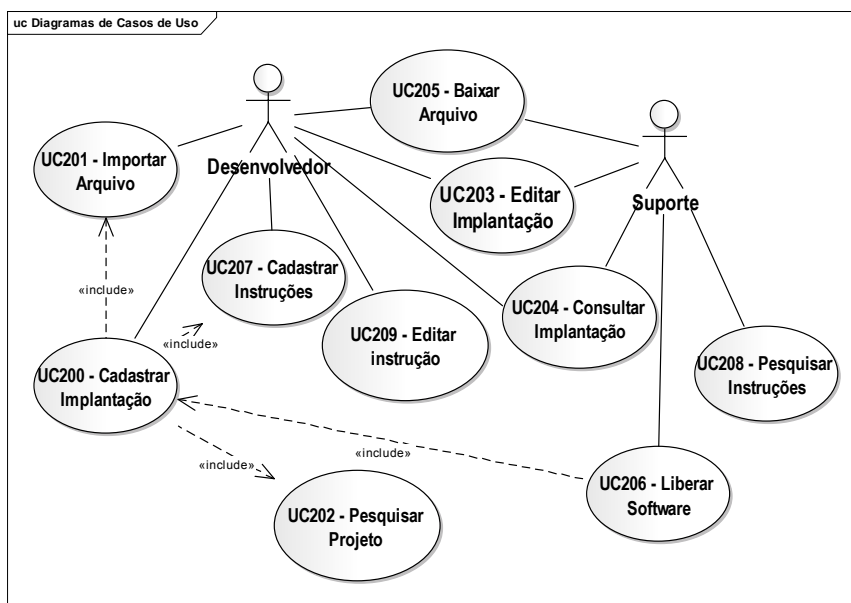


Figura 7. Casos de uso do módulo de Gestão de Implantação

4. Conclusão

Gerenciar bem os projetos dentro das empresas tornou-se não apenas um diferencial competitivo, mas, sobretudo, uma questão de sobrevivência.

O sistema SGPS oferecerá para as organizações e/ou instituições de ensino, todos os processos de gerenciamento de projetos de software desde a sua abertura até a sua implantação. Além disso, o sistema de gestão de projetos ajuda a garantir e promover a qualidade do projeto, reduzir seu tempo, maximizar seus recursos e melhorar a comunicação.

O desenvolvimento do SGPS (Sistema de Gerenciamento de Projeto de Software) está em fase de crescimento, com a gestão de requisitos e de implantação desenvolvidos, aguardando a realização de testes. Além disso, a parte referente ao login do sistema, também está finalizada.

Referências

- Sommerville, Ian. (2006) “Engenharia de Software”, São Paulo, Addison-Wesley, 2006.
- Guedes, G. T. A. (2009) “UML 2 - Uma Abordagem Prática”, São Paulo, Editora Novatec.
- PMI - Project Management Institute. (2004), A Guide Project Management Body of Knowledge. – ANSI/PMI 99-01-2004, Four Campus Boulevard, Newtown Square. USA
- Pressman, Roger. (2004) “Software Engineering-A Practioner's Approach”, McGraw-Hill, 6th edition.
- Sotille, M. (2004) “Gerenciamento de Projetos da Engenharia de Software”. http://www.pmttech.com.br/artigos/Gerenciamento_Projetos_Software.