Hipokratis

Software Requirements Specification

v1.0

2013

Global Chain

10/27/2013

Table of Contents

[Table of Contents 1](#_Toc370659775)

[Table of Diagrams 3](#_Toc370659776)

[1. Introduction 4](#_Toc370659777)

[1.1. Problem Definition 4](#_Toc370659778)

[1.2. Purpose 4](#_Toc370659779)

[1.3. Project Scope 4](#_Toc370659780)

[1.4. Definitions, Acronyms, and Abbreviations 5](#_Toc370659781)

[1.5. References 6](#_Toc370659782)

[1.6. Overview 6](#_Toc370659782)

[2. Overall Description 6](#_Toc370659783)

[2.1. Product Perspective 7](#_Toc370659784)

[2.1.1. System Interfaces](#_Toc370659785) 7

[2.1.2. User Interfaces](#_Toc370659786) 7

[2.1.3. Hardware Interfaces](#_Toc370659787) 7

[2.1.4. Software Interfaces](#_Toc370659788) 7

[2.1.5. Communication Interfaces 7](#_Toc370659789)

[2.1.6. Memory 7](#_Toc370659790)

[2.1.7. Operations](#_Toc370659791) 8

[2.1.8. Site Adaptation Requirements](#_Toc370659792) 8

[2.2. Product Functions](#_Toc370659793) 8

[2.2.1. User Use Cases](#_Toc370659794) 8

[Use Case: Login 9](#_Toc370659795)

[Use Case: Upload Image 9](#_Toc370659796)

[Use Case: Register 10](#_Toc370659797)

[Use Case: Submit 11](#_Toc370659798)

[Use Case: Download 12](#_Toc370659799)

[Use Case: Search 12](#_Toc370659800)

[Use Case: Logout 13](#_Toc370659801)

[Use Case: Edit Profile 13](#_Toc370659802)

[Use Case: Return to Main Page 14](#_Toc370659803)

[2.3. Constraints 14](#_Toc370659804)

[2.4. Assumptions and Dependencies 15](#_Toc370659805)

[3. Specific Requirements 15](#_Toc370659806)

[3.1. Interface Requirements 15](#_Toc370659807)

[3.2. Functional Requirements 15](#_Toc370659808)

[3.2.1. Choose a Scenario 15](#_Toc370659809)

[3.2.2. Upload Image 16](#_Toc370659810)

[3.2.3. Register 16](#_Toc370659811)

[3.2.4. Submit 17](#_Toc370659812)

[3.2.5. Download 17](#_Toc370659813)

[3.2.6. Search 17](#_Toc370659814)

[3.2.7. Logout 18](#_Toc370659815)

[3.2.8. Edit Profile 18](#_Toc370659816)

[3.2.9. Return to Main Page 19](#_Toc370659817)

[3.3. Nonfunctional Requirements 19](#_Toc370659818)

[3.3.1. Performance Requirements 19](#_Toc370659819)

[3.3.2. Design Constraints 19](#_Toc370659820)

[4. Behavioral Model and Description 20](#_Toc370659821)

[4.1. Description for Software Behavior 20](#_Toc370659822)

[4.2. State Transition Diagrams 20](#_Toc370659823)

[5. Planning 21](#_Toc370659824)

[5.1. Team Structure 21](#_Toc370659825)

[5.2. Estimation 21](#_Toc370659826)

[5.3. Process Model 21](#_Toc370659827)

[6. Conclusion 22](#_Toc370659828)

Table of Diagrams

[Diagram 1: Use Cases 5](#_Toc370656468)

[Diagram 2: Login 6](#_Toc370656469)

[Diagram 3: Upload Image 7](#_Toc370656470)

[Diagram 4: Register 7](#_Toc370656471)

[Diagram 5: Submit processed image 8](#_Toc370656472)

[Diagram 6: Search the Database 9](#_Toc370656473)

[Diagram 7: Logout 10](#_Toc370656474)

[Diagram 8: Edit 10](#_Toc370656475)

[Diagram 9: Return to Main Page 11](#_Toc370656476)

[Diagram 10: State Transition 17](#_Toc370656477)

1. Introduction

* 1. Problem Definition

In our country, the radiologists cannot answer the demand because of their workload. Since the radiologists are so busy, sending the patient to them and asking for a roentgen can be a long process. Because of that situation, in some cases, doctors who are not radiologist may have to interpret the radiographic image by themselves. Due to all these reasons, some anomalies in the radiographic images are missed by the doctors.

Human hand contains a lot of little bones and joints. Because of this, it is much easier to make mistakes while interpreting the hand roentgens. For this reason, this problem is especially important in hand roentgens.

* 1. Purpose

This document describes the requirements of a radiographic image processing web application which is called Hipokratis. It aims to describe the required product features, constraints, dependencies and forms basis for design and development phases of the project.

The intended audience of this document is both the users and thedevelopers.

* 1. Project Scope

Everyday, in hospitals, doctors interpret hundreds of radiographic images of different body parts. One of the most complicated parts among them is the human hand. Because there is lots of little bones and joints in the structure of the human hand, there can be mistakes while interpreting these radiographic images. Hipokratis project proposes a solution to speed up the interpretation process of these images and minimize the mistakes.

Hipokratis is a web application to automatically interpret radiographic hand images, detect and point out the anomalies in those images. This application also stores these interpreted images in a database.

* 1. Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| WORD | MEANING |
| SRS | Stands for Software Requirements Specification which completely describes the behavior of a system to be developed. |
| Use Case | Which is a list of steps, typically defining interactions between a role and a system, to achieve a goal. |
| SDLC | Stands for Software Development Life Cycle which is a structure imposed of the development of a software product. |
| Waterfall | Which is a process model that has some strict criterias for the development of an application |
| UC | Stands for Use case |
| PC | Stands for Personal Computer which is used for general purposes. |
| IEEE | Stands for[Institute of Electrical and Electronics Engineers](http://www.google.com.tr/url?sa=t&rct=j&q=ieee&source=web&cd=4&sqi=2&ved=0CFcQFjAD&url=http%3A%2F%2Fwww.ieee.org%2Fportal%2Findex.jsp&ei=2kZvT73cLMj64QTB-7y_Ag&usg=AFQjCNFG7kSY8tJHBrOZpApbxy492HbAuw&cad=rja) |
| Radiography | Radiography is the use of X-rays to view a non-uniformly composed material such as the human body. |
| Radiographic image | Radiographic images are the resulting images of radiography technique. These images are generally used to find anomalies in the bone structure of humans. |
| Radiologist | Radiologists are physicians who use cutting-edge imaging technology to examine organs and tissues inside the body in gentle, noninvasive ways. |
| Web Application | A web-based application is any application that uses a web browser as a client. |

* 1. References
* <http://en.wikipedia.org/wiki/Radiography>
* <http://en.wikipedia.org/wiki/Radiology>
* <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5687039>
* <http://en.wikipedia.org/wiki/Image_processing>
* <http://www.tutorialspoint.com/uml/uml_use_case_diagram.htm>
* <http://www.medikalplus.com/roportaj/98-radyoloji-hizmetinde-nicelik-deil-nitelik-oenemlidir.html>
* <http://www.learningradiology.com/radsigns/radsignspages/S-radsigns.htm>
* <http://medical-dictionary.thefreedictionary.com/radiography>
* <http://docs.oracle.com/javaee/1.4/tutorial/doc/WebApp.html>
  1. Overview

This document describes the software requirements specification for Hipokratis. In the next chapter overall description of the project, main functions,dependencies and constraints are given. In the third chapter specificrequirements of the project, system interfaces, functional requirements,performance and design requirements are given.

1. Overall Description

Hipokratis takes an uploaded radiographic hand image, processes and analyzes it, and returns the result to the doctor. The resulting interpreted image will have the anomalies on it marked and pointed out. The application will allow the user to download the processed images. Hipokratis also stores its resulting data and lets the users to search among the stored images.

* 1. Product Perspective
     1. System Interfaces

On client side the system gets the necessary information by the input devices keyboard and mouse from the user. On server side the system will get the necessary information from the client and output an image file. The client will get the image from the server and display it as output.

* + 1. User Interfaces

The application will be accessed using a webbrowser. In webbrowser window there are several buttons which provides access to functionalites of the application. These buttons and their usage are explained in the Section 2.2.1. User Use Cases.

* + 1. Hardware Interfaces

In order to be able to use the application, a PC which has the basic input devices keyboard and mouse is required.

* + 1. Software Interfaces

There are not any software interfaces to other components of the software system. The system does not include any component that is purchased or reused from another application. All components of the system will be developed newly.

* + 1. Communication Interfaces

The client and server components of the application communicates over http protocol.

* + 1. Memory

There are not any required limitations for memory. Memory capabilities of any system would be enough to run this software.

* + 1. Operations

The user must open a web browser program and navigate to the web site of the Hipokratis Application.

* + 1. Site Adaptation Requirements

There are not any adaptation requirements.

* 1. Product Functions
     1. User Use Cases

The User has the following sets of use cases:

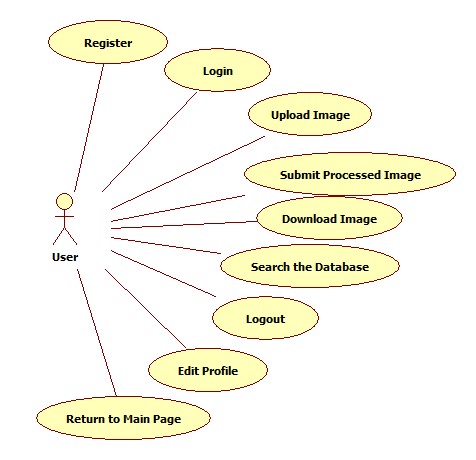


Diagram 1: Use Cases

Use Case: Login

**Diagram**

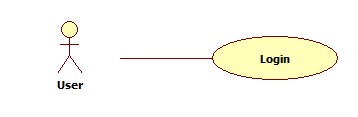


Diagram 2: Login

**BriefDescription**

The user clicks on the login button to be logged into the application using username and password.

**Initial Step-By-Step Description**

Before this use case can be initiated, the user must have already entered the web page of the application.

1. Application provides a login button and two editable text boxes for user to enter username and password.

2. The user enters the username and the password into the text boxes provided and clicks on the login button.

3. If the username and password checks out by the application, the user will become successfully logged in.

4. If the username and password does not checked out application will show a warning message.

Use Case: Upload Image

**Diagram**

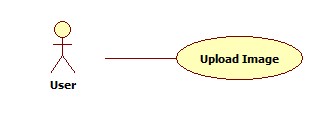


Diagram 3: Upload Image

**BriefDescription**

The user clicks on the upload button and browses for the image to be uploaded. When the image is selected and uploaded, it is processed by the application in the background and the resulting image is returned back to the user.

**Initial Step-By-Step Description**

Before this use case can be initiated, the user must have already entered the web page of the application and successfully logged into the system.

1. Application provides anupload button and user clicks that button.

2. Application pops up a window to browse for the image file to be uploaded.

3. The user finds and selects an image.

4. Application uploads the selected image to the server and processes it.

5. The resulting processed image is shown on the web page.

Use Case: Register

**Diagram**

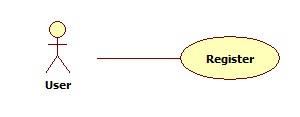


Diagram 4: Register

**BriefDescription**

The user will register to the system by providing a username and password and using an access code given by the system administrators.

**Initial Step-By-Step Description**

Before this use case can be initiated, the user must have already entered the web page of the application.

1. The user clicks the register button.

2. Application provides 4 editable text boxes for username, password, retype password and access code fields. It also provides Register and Cancel buttons.

3. If the user clicks the cancel button at any time, application returns to the starting page.

4. The user enters the related information into the provided text boxes.

5. The user clicks the register button.

6. If all the fields check out, the user will become registered.

7. If any field does not check out, the application will give a warning message.

Use Case: Submit

**Diagram**

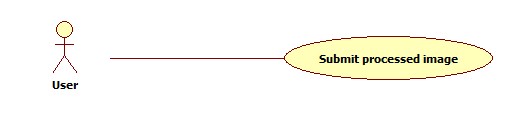


Diagram 5: Submit processed image

**BriefDescription**

The user will enter a diagnoses and submit the processed image into the database.

**Initial Step-By-Step Description**

Before this use case can be initiated, the user must have already entered the web page of the application, logged into the system and uploaded a valid image.

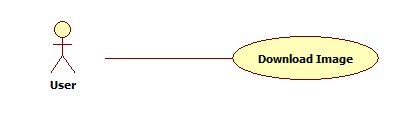
1. Application provides an editable text box for diagnoses.

2. The user enters a diagnoses into the provided text box.

3. The user clicks the submit button

Use Case: Download

**Diagram**



**BriefDescription**

The user downloads the processed image.

**Initial Step-By-Step Description**

Before this use case can be initiated, the user must have already entered the web page of the application, logged into the system and uploaded a valid image.

1. The user clicks the download button .

2. The image is downloaded to user’s computer

Use Case: Search

**Diagram**

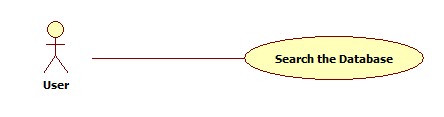


Diagram 6: Search the Database

**BriefDescription**

The user searches the database with some keywords.

**Initial Step-By-Step Description**

Before this use case can be initiated, the user must have already entered the web page of the application, logged into the system.

1. The application provides an editable text box for keywords.

2. The user enters the desired keywords.

3. The user clicks the search button .

4. The application finds and lists the images from database according to entered keywords.

5.The user clicks on an image from the list to view it.

Use Case: Logout

**Diagram**

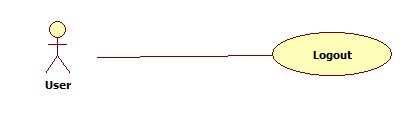


Diagram 7: Logout

**BriefDescription**

The user logs out from the system.

**Initial Step-By-Step Description**

Before this use case can be initiated, the user must have already entered the web page of the application, logged into the system.

1. The user clicks on the logout button.

2. The user becomes logged out of the system.

Use Case: Edit Profile

**Diagram**

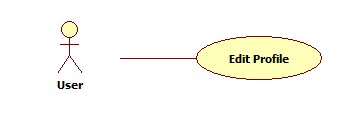


Diagram 8: Edit

**BriefDescription**

The user changes the username and password.

**Initial Step-By-Step Description**

Before this use case can be initiated, the user must have already entered the web page of the application, logged into the system.

1. The user clicks on the edit profile button.

2. The application provides 4 editable text boxes for username, old password, new password, new password retype.

3. The user clicks the save button.

4. If the operation is successful, changes are saved.

5. If the operation fails, application gives a warning message.

Use Case: Return to Main Page

**Diagram**

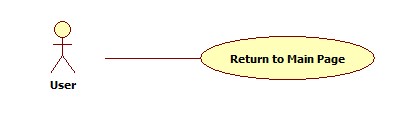


Diagram 9: Return to Main Page

**BriefDescription**

The user returns to the main page from one of the other pages. Main page is the page where the user is directed after a successful login operation.

**Initial Step-By-Step Description**

Before this use case can be initiated, the user must have already entered the web page of the application, logged into the system. The user must have executed the search or upload image operations.

1. The user clicks on the return button.

2. The user returns to the main page.

* 1. Constraints

Only the registered users can use the system functionality of the application.

* 1. Assumptions and Dependencies

There is no certain dependencies that may affect the requiements stated in this SRS document.

1. Specific Requirements
   1. Interface Requirements

File system inteface of the operating system ,on which the application is running, is required during the selection of image file. Browser interface is required for interacting with the application.

* 1. Functional Requirements

Description of each function of the software is supplied below.

* + 1. Choose a Scenario

|  |  |
| --- | --- |
| Use Case Name | UC.01 Login |
| Xref | Section 2.2.1.1 Use case: Login |
| Actor | User |
| Trigger | The user clicks the login button |
| Precondition | The user must have already entered the web page of the application. |
| Basic Path | 1. Application provides a login button and two editable text boxes for user to enter username and password.  2. The user enters the username and the password into the text boxes provided and clicks on the login button.  3. If the username and password checks out by the application, the user will become successfully logged in.  4. If the username and password does not checked out application will show a warning message. |
| Postcondition | The user is logged in |

* + 1. Upload Image

|  |  |
| --- | --- |
| Use Case Name | UC.02Upload Image |
| Xref | Section 2.2.1.2 Use case: Upload Image |
| Actor | User |
| Trigger | The user clicks the upload button |
| Precondition | The user must have already entered the web page of the application and successfully logged into the system. |
| Basic Path | 1. Application provides an upload button and user clicks that button.  2. Application pops up a window to browse for the image file to be uploaded.  3. The user finds and selects an image.  4. Application uploads the selected image to the server and processes it.  5. The resulting processed image is shown on the web page. |
| Postcondition | The resulting processed image is shown on the web page. |

* + 1. Register

|  |  |
| --- | --- |
| Use Case Name | UC.03 Register |
| Xref | Section 2.2.1.3 Use case: Register |
| Actor | User |
| Trigger | The user clicks the register button |
| Precondition | The user must have already entered the web page of the application |
| Basic Path | 1. The user clicks the register button.  2. Application provides 4 editable text boxes for username, password, retype password and access code fields. It also provides Register and Cancel buttons.  3. If the user clicks the cancel button at any time, application returns to the starting page.  4. The user enters the related information into the provided text boxes.  5. The user clicks the register button.  6. If all the fields check out, the user will become registered.  7. If any field does not check out, the application will give a warning message. |
| Postcondition | The user is registered. |

* + 1. Submit

|  |  |
| --- | --- |
| Use Case Name | UC.04 Submit |
| Xref | Section 2.2.1.4 Use case: Submit |
| Actor | User |
| Trigger | The user clicks the submit button |
| Precondition | The user must have already entered the web page of the application, logged into the system and uploaded a valid image. |
| Basic Path | 1. Application provides an editable text box for diagnoses.  2. The user enters a diagnoses into the provided text box.  3. The user clicks the submit button |
| Postcondition | The image is submitted into the database. |

* + 1. Download

|  |  |
| --- | --- |
| Use Case Name | UC.05 Download |
| Xref | Section 2.2.1.5 Use case: Download |
| Actor | User |
| Trigger | The user clicks the download button |
| Precondition | The user must have already entered the web page of the application, logged into the system and uploaded a valid image. |
| Basic Path | 1. The user clicks the download button .  2. The image is downloaded to user’s computer |
| Postcondition | The image is downloaded to user’s computer |

* + 1. Search

|  |  |
| --- | --- |
| Use Case Name | UC.06 Search |
| Xref | Section 2.2.1.6 Use case: Search |
| Actor | User |
| Trigger | The user clicks the search button |
| Precondition | The user must have already entered the web page of the application, logged into the system. |
| Basic Path | 1. The application provides an editable text box for keywords.  2. The user enters the desired keywords.  3. The user clicks the search button .  4. The application finds and lists the images from database according to entered keywords.  5.The user clicks on an image from the list to view it. |
| Postcondition | Images from database are filtered according to keywords and the resulting images are listed. |

* + 1. Logout

|  |  |
| --- | --- |
| Use Case Name | UC.07 Logout |
| Xref | Section 2.2.1.7 Use case: Logout |
| Actor | User |
| Trigger | The user clicks the logout button |
| Precondition | The user must have already entered the web page of the application, logged into the system. |
| Basic Path | 1. The user clicks on the logout button.  2. The user becomes logged out of the system. |
| Postcondition | The user is logged out of the system. |

* + 1. Edit Profile

|  |  |
| --- | --- |
| Use Case Name | UC.08 Edit Profile |
| Xref | Section 2.2.1.8 Use case: Edit Profile |
| Actor | User |
| Trigger | The user clicks the edit profile button. |
| Precondition | The user must have already entered the web page of the application and logged into the system. |
| Basic Path | 1. The user clicks on the edit profile button.  2. The application provides 4 editable text boxes for username, old password, new password, new password retype.  3. The user clicks the save button.  4. If the operation is successful, changes are saved.  5. If the operation fails, application gives a warning message. |
| Postcondition | Username and password are successfully changed. |

* + 1. Return to Main Page

|  |  |
| --- | --- |
| Use Case Name | UC.09 Return to Main Page |
| Xref | Section 2.2.1.9 Use case: Return to Main Page |
| Actor | User |
| Trigger | The user clicks the edit profile button. |
| Precondition | The user must have already entered the web page of the application, logged into the system. The user must have executed the search or upload image operations. |
| Basic Path | 1. The user clicks on the return button.  2. The user returns to the main page. |
| Postcondition | Application returns to the main page. |

* 1. Non-functional Requirements
     1. Performance Requirements

Any number of users will be able to use the application simultaneously. The maximum amount of time for image to be processed will be 30 seconds. The upload and download time of the processed image depends on the network capabilities of the user’s computer.

* + 1. Design Constraints

There are no constraint on the programming language used in the implementation of this project. The application will work on any computer that has a network connection and an installed web browser. The application will work on MAC, Linux and Windows operating systems.

The application will have a registration system that restricts the access to the application system. The database will keep itself safe and consistent.

Unless there is a need for a server maintenance, the web page of the application will always be accessible.

1. Behavioral Model and Description
   1. Description for Software Behavior

* Beginning
* Register (if not registered before)
* Login
* Upload image
* Download the result (optional)
* Submit the processed image (optional)
* End
  1. State Transition Diagrams

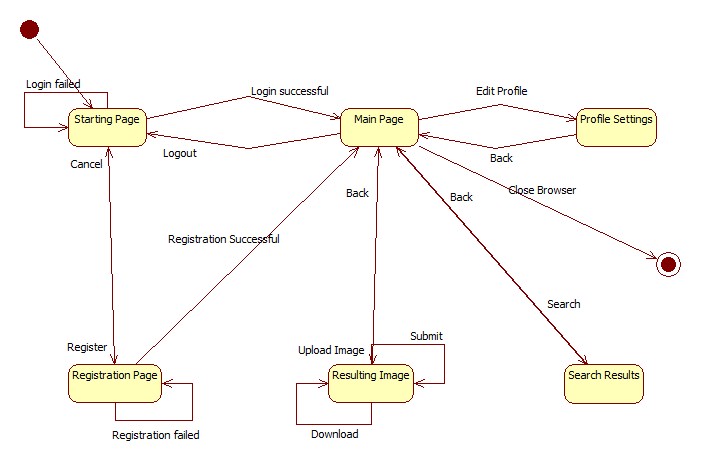


Diagram 10: State Transition

1. Planning
   1. Team Structure

Team members are,

* Barış KENİŞ
* Ahmet KORKMAZ
* Erkan ONAT
* Oğuz ÖRCÜN

Barış Keniş and Oğuz Örcün are responsible for image processing. Ahmet Korkmaz and Erkan Onat web and database design.

* 1. Estimation
* Submission of the SRS : 30 October 2012
* Submission of the SDD : 24 November 2012
* Submission of the STD : TBA
* Demo of the Project : TBA
* Submission of the Final Report : TBA
  1. Process Model

The SDLC of waterfall is used in this application. The lifetime of a waterfall model is described below:

* Requirement Specification
* Software Design
* Implementation and Integration
* Testing
* Deployment
* Maintenance

1. Conclusion

This SRS is prepared in order to explain the full design and process for the application. The aim was to make users and developers keen on the usage of Hipokratis Web Application.