



Deggendorf Institute of Technology

**Functional Specifications
Of The Project**

“COSMOS INFINITY BLOG”

(Advanced Programming Techniques)

Professor

Dr. Andreas Fischer

INITIAL REPORT

Objective:

The aim of this project is to get hands on practice on latest programming languages and familiarize ourselves with advanced programming techniques and also working in group of multiple developers.

Introduction:

The blog will be about the cosmos and its vastness. It will allow the authors to write about the universe and the solar system and also about other galaxies.

Collaborators:

Names	Matriculation
Willian Condor	00096685
Faizan Mahmood	00096480
Jawad Khan	00097005
Rohaib Hassan	00010997

Implementation Requirements:

The project will be divided by tasks detailed in the project responsibility section below, we have to create algorithms, codes, scripts, html files to meet the requirements and give a solution for the needs.

The Project approaches to build a blogging software that met the whole requirements state in the Section 1, the final user could be anyone, user will create articles about any topic they want after proper registration. The programing language could be

Python due to advantage of having different web frameworks available. In addition, Python run on Microsoft Windows, Linux and Mac OS, it is a cross-platform that is intended to run on low-end hardware systems.

Python is regarded as one of the best and easiest programming languages for beginners. It's a free and open source language that's known for its object-oriented programming style. Programmers can express concepts in lesser lines of code.

Python is now developed by Google, is used for scientific computing, data mining and machine learning. However, it is a very versatile language that allows to create web applications. We propose to use the Django framework for improving the developing time and reduce the requirements of installing web servers or additional packages – Python emulate the services of the web server through the Django Framework-.

Core of the project: Python + Django Framework

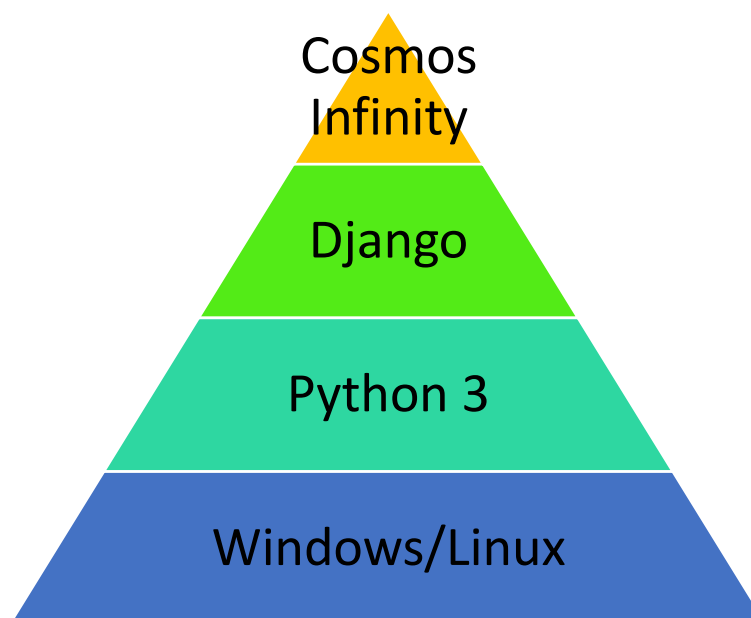
OS supported: Windows, Linux

Technologies for front-end: HTML, CSS, Bootstrap, and Javascript.

Database: SQL Database.

Technologies for Back-end: Python, Django

Targets systems: Linux and Windows



Description for fulfilling the requirements

Task 1. Blog with several pages and a main pages reachable through hyperlinks.

When a new article is created in the blog, the software will assign an ID to the new article page. The articles could be accessed through url: domain/article_ID=<id>

Task 2. Levels access for different users

The blog software gives different permissions for registered user through username and password. The different permissions are granted if the username (email) and password are entered correctly.

Task 3. Control comment feature

When a new comment is created, the system assigns an ID. The software checks for new comments and if the comment was created for a guest user. It is not allowed to display without an administrator permission.

Task 4. Login Feature

The page will have a special space (container) into the web page for login. The authentication algorithm will be performed by the back-end.

Task 5. Sign up features

The blog will have a special web page with a form for registration as a new user. It is necessary to provide email, password etc. to create a new account.

Task 6. Structure of the article layout

We will create a web template where the information will be displayed. The system will render the template and access to the database for retrieving the necessary information.

Task 7. Upload picture features

If the article has a picture, the system assign a <picture_ID> for storing and accessing the image.

Task 8. Structure of the main page layout

We will create a web template for the main page where the information will be displayed. The system will render the template and access to the database for retrieving the necessary information.

Task 9. And Task 10. Keyword indexing system and Sorting feature by date and keyword

The articles will be indexed by ID, tags and dates. When the user searches an article. The software can sort this information through a SQL request.

Task 11. Implements measurements of load time

The system will store data about rendering-times and performance.

Task 12. Test for logging features

The system will check the login information, whether is correct or not.

Task 13. Test for URL endpoint

The system will validate the URL endpoint through session for different user.

Task 14. Permission test for account

Admin will allow the user to login in the system. Whenever a new account being created admin will accept that user then he will be able to login.

Task 15. Test for user inputs

The system validates the information entered by the user before storing in the system.

Task 16. Default account for administration

After installation the platform will create a default account for administration

Task 17. Database deployment

The data schemes of different tables will be designed for meeting the requirements and needs of the work team.

Project Responsibilities:

1. Faizan Mahmood

Matriculation No: 00096480

Back-end: Develop Functional Requirements of our Application using Python and Django.

2. Jawad Khan

Matriculation No: 0097005

Front-end: Develop basic web page content and control the visual layout.

3. Willian Condor

Matriculation No: 00096685

Security: Implement appropriate controls to make sure security requirements are enforced.

4. Rohaib Hassan

Matriculation No: 00096646

Testing: Develop an appropriate testing suite to test functionality developed