**AICTE BUILD A THON 2023**

**24.07.2023 – 04.08.2023**

**Project Report**

**under**

**the title**

**Smart Assignment Management System**

**Submitted by**

**Dr. R GUNABALAN**

**Professor**

**School of Electrical Engineering**

**Vellore Institute of Technology – Chennai**



**AUGUST 2023**

1. **Introduction**

Choice based credit system (CBCS) is implemented presently in all higher educational institutions as per AICTE regulations. The CBCS system is learner centric. Internal assessment is one of the most important components in the present higher education system. The weightage given for continuous internal assessment is litter higher than the final assessment in order to engage students continuously. It improves their self-learning capability, critical thinking, decision making leadership quality etc. The evaluation process is a challenging task for faculty members. In this project work, a simple autonomous evaluation process is implemented using cloud app.

1. **Objectives**

The main objectives of the project work are to:

1. Leverage web development technologies to create an intelligent platform that automates assignment creation, submission, grading, and feedback, thereby improving the efficiency and effectiveness of the assignment process.
2. Enhance the assignment management process, promote collaboration, and improve educational outcomes.
3. Streamlines administrative tasks, enhances student-teacher communication, and facilitates a more efficient and engaging learning experience.
4. **Literature survey**

Every project or work or event involves series of organized activities for the success of the project/work/event. Project management system helps to identify the issues and provides a solution. In higher educational institutions, the assignment submission is an important task for students for internal assessment system. The steps flowed by the instructor (professor) are: Problem identification, description, evaluation rubrics, deadline, submission link etc. After submission of the assignment by students, the instructor evaluates the content as per rubrics and uploads the marks in the learning management system (LMS) for online submission. The steps followed by students are: Understanding the problem statement, following the rubrics, content writing and submission. After evaluating the assignment by instructor, queries raised by the students would be clarified by the instructor.

LMS is used in most of the higher educational institutions for assignment submission and evaluation. The limitation is student can access the assignment within the campus. Web based assignment submission will solve the issue. Student can access the assignment anywhere, anytime with necessary internet connection.

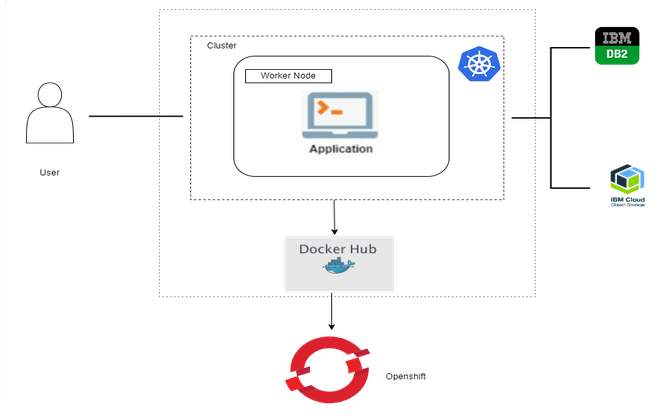
The major advantages of web based assignment submission system are:

* Accessible anywhere
* Reduce paper handling
* Easy communication
* Online monitoring

A study on task management system was discussed in detail for easy and effective task management [1]. A web based assignment system was developed by researchers using agile technology [2]. The process was explained in detail and the merits of online submission system were discussed.

1. **Block diagram**

The block diagram of the smart assignment management system interface is shown in Figure 1. The required data for the smart assignment system is created by the user. The coding is written in spyder and ‘html’ pages. Watson assistant is used for creating the chatbot. The details were stored in IBM cloud using DB2 in IBM resources. The entire coding is converted into a single image using docker desktop. Docker hub and docker desktop are used for pulling and pushing the image. The Hedhat is an alternate option for implementation. The details procedures are needed for image creation, pulling and pushing the image.



**Figure 1 block diagram for smart assignment management system interface**

1. **Methodology**

**5.1 Necessary steps:**

* create HTML pages
* create object storage
* create flask file
* deployment in docker hub
* deployment in redhat
* deployment in Github

**5.2 Requirements**

To build the model, the following packages are required:

Spyder tool for writing the coding in python language

HTML coding for website integration

Db2 in IBM cloud

IBM Watson assistant

Docker hub

Docker desktop

Redhat etc.

.  
**5.3 Algorithm steps**

* Registration
* login
* Get the instructions from chatbot
* Create Whatson assistant
* Web integrations
* Docker window and Docker desktop for image
* creating single image in docker
* pulling and pushing image
* Redhat implementation
* Github implementation

### Results and Discussion

### 6.1 Smart Assignment System with Docker Desktop and Docker Hub

### The model is built using python coding, written in spyder notebook. The python coding and html coding are used to execute the program. Figure 2 and Figure 3 show the screenshot of the web page created for registration and login for assignment submission system. The student details are available after entry in the registration page. The application was created with user interaction with the support from IBM Watson assistant. The created application was converted into image using docker desktop and docker hub. The application was executed in the docker desktop itself using containers. The docker image available in the docker desktop was pulled into docker hub. Figure 4 to Figure 18 indicates the various implementation steps involved to create webpage for the given coding. The application (image) is pulled into to docker hub at the end. The image is pushed into docker hub and finally pulled back to docker desktop from docker hub. Figure 17 and Figure 18 indicate the screenshot of the push and pull command.

### 

### Figure 2 Student registration form for assignment submission

### 

### Figure 3 student login page for assignment submission system

### 

### 

### 

### Figure 5 Students assignment submission system with profile details

### 

### Figure 6 Student details in Db2

### Watson Assistant

### 

### Figure 7 Watson assistance for chatbot creation

### 

### Figure 8 Chatbot for assignment submission system

### 

### Figure 9 Chat bot is active in the smart assignment system

### 

### Figure 10 Web integration in Watson assistant

### 

### Figure 11 Docker desktop image creation process

### 

### Figure 12 Docker image created

### 

### Figure 13 Docker image creation in local server

### 

### Figure 14 User login page in the local server

### 

### Figure 15 Different docker images created in docker hub

### 

### Figure 16 Docker image is pushed to docker hub

### 

### Figure 17 Pushed docker image in docker hub

### 

### Figure 18 Pulled docker image from docker hub to docker desktop

### 5.2 Smart Assignment System with Redhat

### An account is created in redhat to deploy the application created in docker desktop and docker hub. After creating the account, the log in page is shown in Figure 19. The image and application deployment are shown in Figure 20 and Figure 21. After deployment, the application window is shown in Figure 22.

### 

### Figure 19 Login page in redhat

### 

### Figure 20 Image deployment in redhat

### 

### Figure 21 Application deployment in redhat

### 

### Figure 22 After application deployment with URL

### 5.3 Smart Assignment System with Github in Redhat

### An account is created in Githubt to deploy the application in redhat. After creating the account, depository is created in Github as shown in Figure 23. The redhat import and application deployment are shown in Figure 24 and Figure 25.

### 

### Figure 23 Git Hub repository

### 

### Figure 24 Redhat import

### 

### Figure 25 Github application deployment in redhat

### 5.4 Smart Assignment System with YAML in Redhat

### The YAML code is copied in redhat environment. The YAML code is created in redhat. While execution error was created and the process is stopped further. The YAML execution is shown in Figure 26.

### 

### Figure 26 YAML application deployment in redhat

### Conclusion

### The smart assignment system is created and deployed in Github, docker hub and redhat. The procedural steps are explained in detail. It is a good experience and learnt about cloud app development with redhat.

### Acknowledgment

### Author thank AICTE and smartinternz trainers for their valuable input to complete the training successfully and complete the project work.

### References

1. N. S. Jyothi and A. Parkavi, "A study on task management system," 2016 International Conference on Research Advances in Integrated Navigation Systems (RAINS), Bangalore, India, 2016, pp. 1-6, doi: 10.1109/RAINS.2016.7764421.
2. Muhammad E R, Zailan A and Darshini A, “Automated Academic Assignment Submission and Management System: Proposed Design Using Agile Methodology”, Proceedings of the 3rd International Conference on Integrated Intelligent Computing Communication & Security (ICIIC 2021), Atlantis Highlights in Computer Sciences, volume 4, 2021.

### Recorded video lecture by smartinetrnz