BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

FIRST Semester 2022-23

SEWP ZG628T DISSERTATION

Dissertation Outline

BITS ID No. 2018HW70294 **Name of Student:** AKHIL GUPTA

E-mail ID of the student: 2018hw70294@wilp.bits-pilani.ac.in

Name of Supervisor: VIKRAM KUMAR

Designation of Supervisor: Administrator - ICORE- CIS

Qualification and Experience: B.Tech. (CSE) with 7.1 Year Experience

E- mail ID of Supervisor: vikram.kumar19@wipro.com
vikram01234@gmail.com

Title of Dissertation: GitHub Org Migration: Self-hosted to Cloud (Automation)

Name of First Examiner: MONI KUMARI

Designation of First Examiner: Project Engineer - IDEAS-APPS ENGG & MODERNIZATION

Qualification and Experience: B.Tech. (CSE) with 4.1 Year Experience

Email ID of First Examiner: moni.kumari@wipro.com
nrf10d11@gmail.com

Name of Second Examiner: SANAPALA ALEKHYA

Designation of Second Examiner: Project Engineer EURP UK & Ireland WT01

Qualification and Experience: B.Tech. with 4.3 Year Experience

Email ID of Second Examiner: sanapala.alekhya@wipro.com
Alekhya.sanapala2344@gmail.com

Supervisor's rating of the Technical Quality of this Dissertation Outline

EXCELLENT / GOOD / FAIR/ POOR (Please specify): ________________________

Supervisor's suggestions and remarks about the outline:

Gupta

(Signature of Student) (Signature of Supervisor)

Date: ----------- Date: --------------

Body of the outline

1. **Discussion on the chosen topic**:

- **Introduction:** Version control tools like Git, Concurrent Version System (CVS), Mercurial, and many others are widely used in the IT development area. Such tools offer to streamline the development process and keep a history of all changes within a code. So, if a developer makes a mistake, then it can be undone and fixed by comparing the new code with a previous version. Version Control Software (VCS) also known as Source Code Management (SCM) tools or Revision Control System (RCS) is a great fit for any web development company around the globe.

- **Scope of work:** With the increase in demand for software, codebase data is gradually increasing on servers. To manages such data, the IT infrastructure needs to expand the existing data center, which involves large CapEx and OpEx. To overcome this challenge cloud is the ultimate solution. Still migrating the data from self-hosted VCS to the cloud is another big challenge as this involves various categories of data for migration like code history, pull requests, releases, issues, etc.
  Using the planned automation "GitHub Org Migration: Self-hosted to Cloud" the admin team can easily migrate the GitHub Org to the cloud with reduced overhead.

- **Background of previous work done:** Migrating any GitHub organization from self-hosted to enterprise cloud requires various manual activities to be performed by the admin team, few are mentioned below:
  - Create an empty org on GitHub enterprise cloud.
  - Provide enterprise license to users (Providing access).
  - Adding users with the specific role (owner/member) in the new org.
  - Create a migrator file for repositories from self-hosted GitHub and import it into GitHub enterprise.

- **Methodology:** To reduce the overhead of a series of manual activities can be automated, as suggested below:
  - Input the name of the org that exists in the self-hosted GitHub. A Python script can be used for gathering the active owner and members of entered organizations.
  - Microsoft GraphQL (https://docs.microsoft.com/en-us/graph/api/overview?view=graph-rest-1.0) can be used to validate users' active status and provide the license to access the GitHub enterprise cloud using an automation script.
  - Using APIs provided by GitHub (https://docs.github.com/en/rest/guides/getting-started-with-the-rest-api) new organization can be created on the GitHub enterprise cloud. Members can be assigned to the created org with respective roles using the same.
  - A prepared migrator file can be imported to https://eci.github.com/ for successful migration or repositories. Documentation can be found at http://git.io/vtLRM.

- **Benefits expected from the work:** Using the planned automation, manual activities such as creating org, verifying users' active status, and providing licenses for user access get reduced for the admin team. A final report containing logs from all activities generated by the script can be reviewed by the admin team. This will come out as the ultimate solution while performing bulk migration of organizations.

2. Detailed Plan of Work (for 16 weeks)

The plan of work is specified in the table given below:

Serial Number of Task	**Tasks or subtasks to be done** (be precise and specific)	Planned duration in weeks	Specific Deliverable in terms of the project
1	Review documentation and API identification for self-hosted GitHub.	02	Understand the workflow of self-hosted GitHub.
2	API identification for GitHub enterprise cloud	01	Connection to fetch required data from GHEC.
3	Accessibility to the Active Directory using tokens and secrets. And analyze Microsoft APIs.	02	Connection to Active Directory for verifying users and license procurement.
4	Develop a script to get users' detail of input org name	02	Get the list of org members and owners for further usage.
5	Code to filter active users in Azure Active Directory and user licensing	03	Provide access to the active users for GitHub enterprise cloud.
6	Create a script for new org creation and member addition on GitHub enterprise cloud	03	A new org will be created, and role (owner/member) assigned to users.
7	Created migrator file (.tar.gz) and required access tokens for authorization.	02	Data Migration file created, which can be imported to https://eci.github.com/.
8	Integrate created scripts and perform system testing.	01	Completion of the final project.