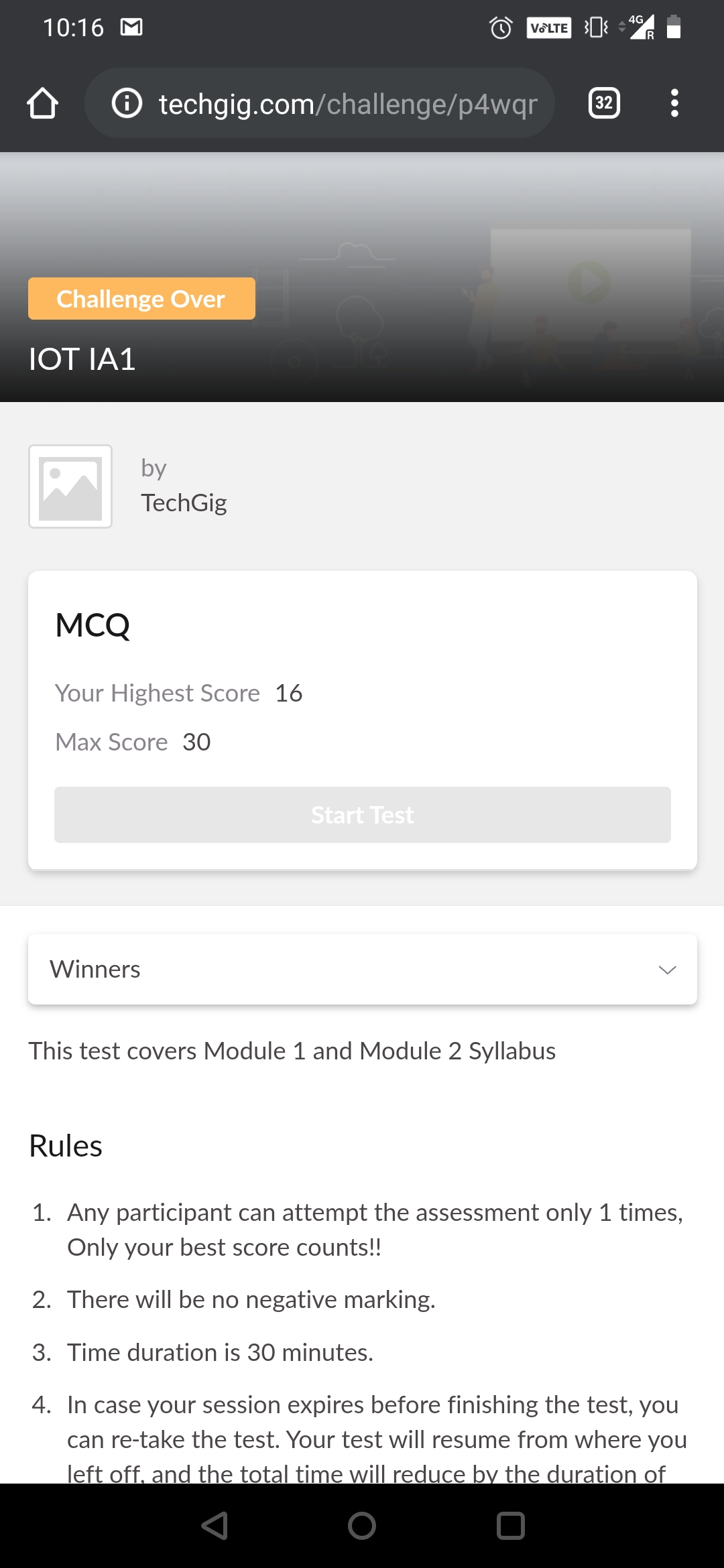
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **20/05/2020** | | | | **Name:** | **Samrin banu** | |
| **Sem & Sec** | **8th B** | | | | **USN:** | **4AL16CS112** | |
| **Online Test Summary** | | | | | | | |
| **Subject** | | **IOT** | | | | | |
| **Max. Marks** | | **30** | | **Score** | | **16** | |
| **Certification Course Summary** | | | | | | | |
| **Course** | **Introduction To Hadoop** | | | | | | |
| **Certificate Provider** | | | **Great Learning Academy** | **Duration** | | | **30 mins** |
| **Coding Challenges** | | | | | | | |
| **Problem Statement:** | | | | | | | |
| **Status: Solved** | | | | | | | |
| **Uploaded the report in Github** | | | | **yes** | | | |
| **If yes Repository name** | | | | **samrinbanu** | | | |
| **Uploaded the report in slack** | | | | **yes** | | | |

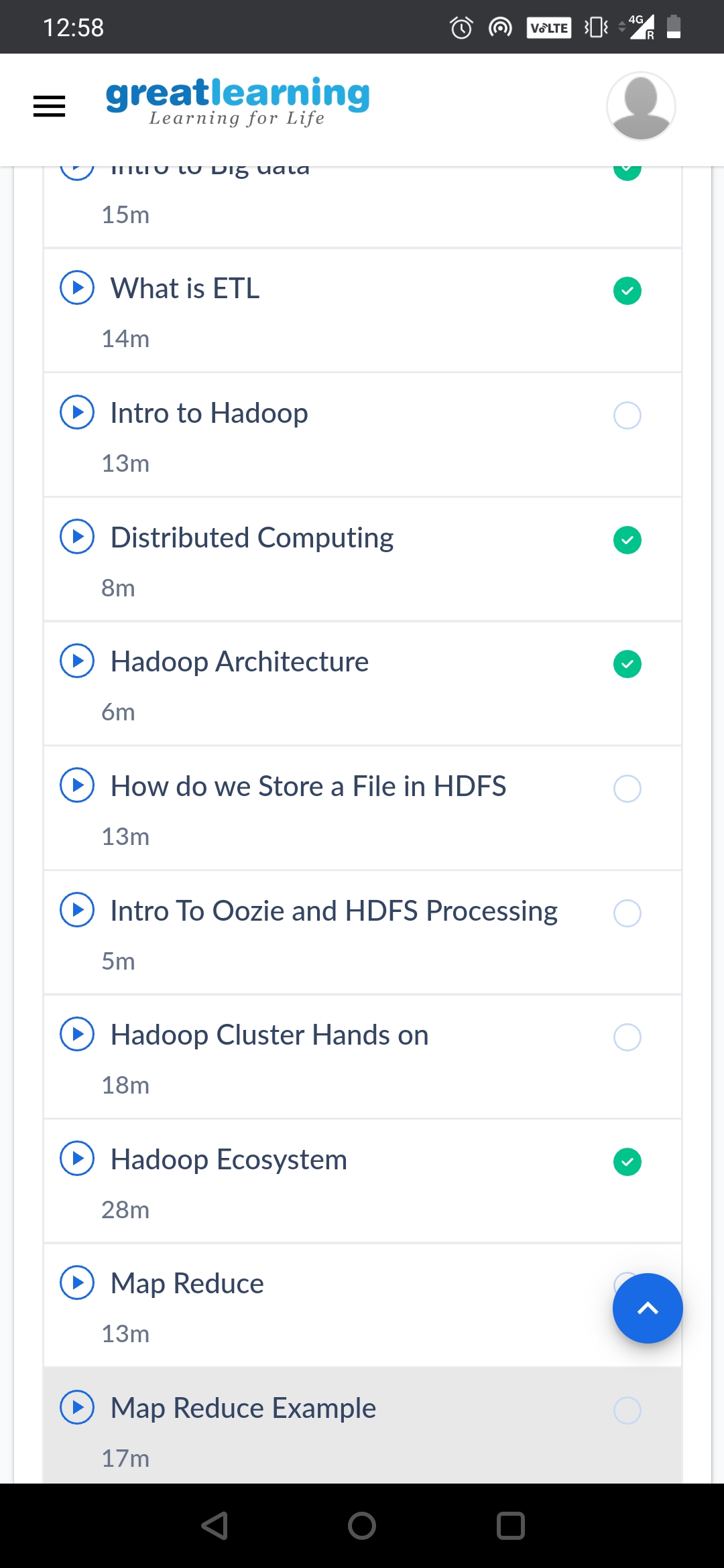
**Online Test Details:**



**Certification Course Details**:

**Hadoop ecosystem:-**

The Hadoop ecosystem includes multiple components that support each stage of Big Data processing. Flume and Sqoop ingest data, HDFS and HBase store data, Spark and MapReduce process data, Pig, Hive, and Impala analyze data, Hue and Cloudera Search help to explore data. Oozie manages the workflow of Hadoop jobs.



**Coding Challenges Details**:

**program1:**

**struct Node**

**{**

**int data;**

**struct Node\* next;**

**};**

**pointer to the new head node. /**

**struct Node reverse (struct Node head, int k)**

**{**

**struct Node current = head;**

**struct Node next = NULL;**

**struct Node prev = NULL;**

**int count = 0;**

**while (current != NULL && count < k)**

**{**

**next = current->next;**

**current->next = prev;**

**prev = current;**

**current = next;**

**count++;**

**}**

**if (next != NULL)**

**head->next = reverse(next, k);**

**return prev;**

**}**

**void push(struct Node\*\* head\_ref, int new\_data)**

**{**

**struct Node\* new\_node =**

**(struct Node\*) malloc(sizeof(struct Node));**

**new\_node->data = new\_data;**

**new\_node->next = (\*head\_ref);**

**(\*head\_ref) = new\_node;**

**}**

**void printList(struct Node \*node)**

**{**

**while (node != NULL)**

**{**

**printf("%d ", node->data);**

**node = node->next;**

**}**

**}**

**int main(void)**

**{**

**struct Node\* head = NULL;**

**push(&head, 8);**

**push(&head, 7);**

**push(&head, 6);**

**push(&head, 5);**

**push(&head, 4);**

**push(&head, 3);**

**push(&head, 2);**

**push(&head, 1);**

**printf("\nGiven linked list \n");**

**printList(head);**

**head = reverse(head, 2);**

**printf("\nReversed Linked list \n");**

**printList(head);**

**return(0);**