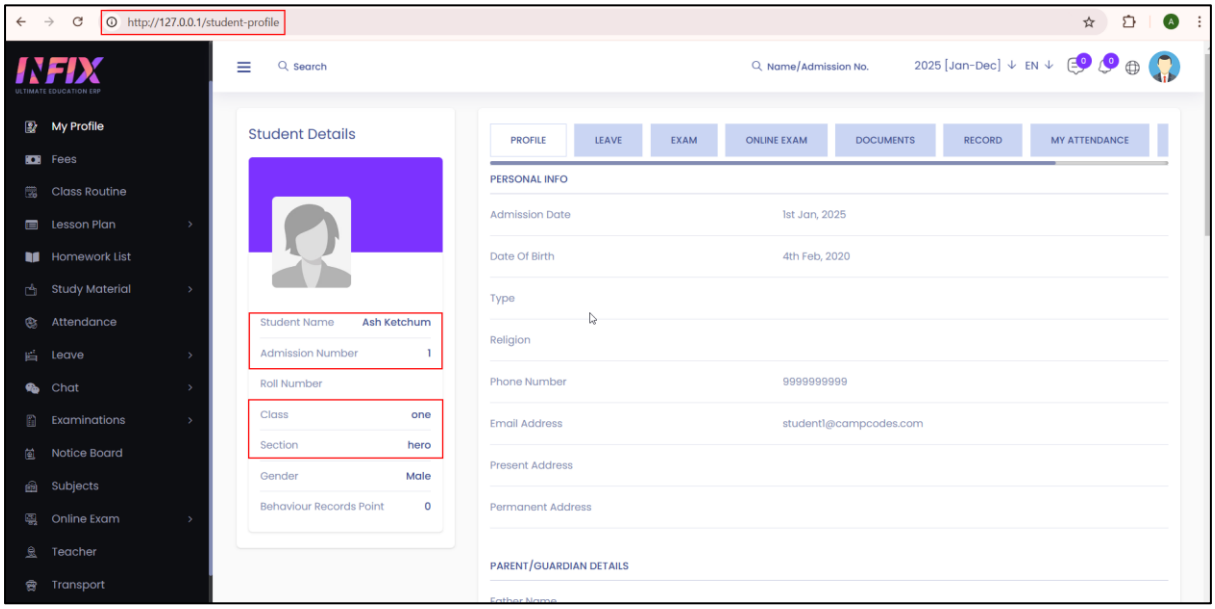
Insecure Direct Object Reference (IDOR) – All Student Homework Downloadable

Observation:

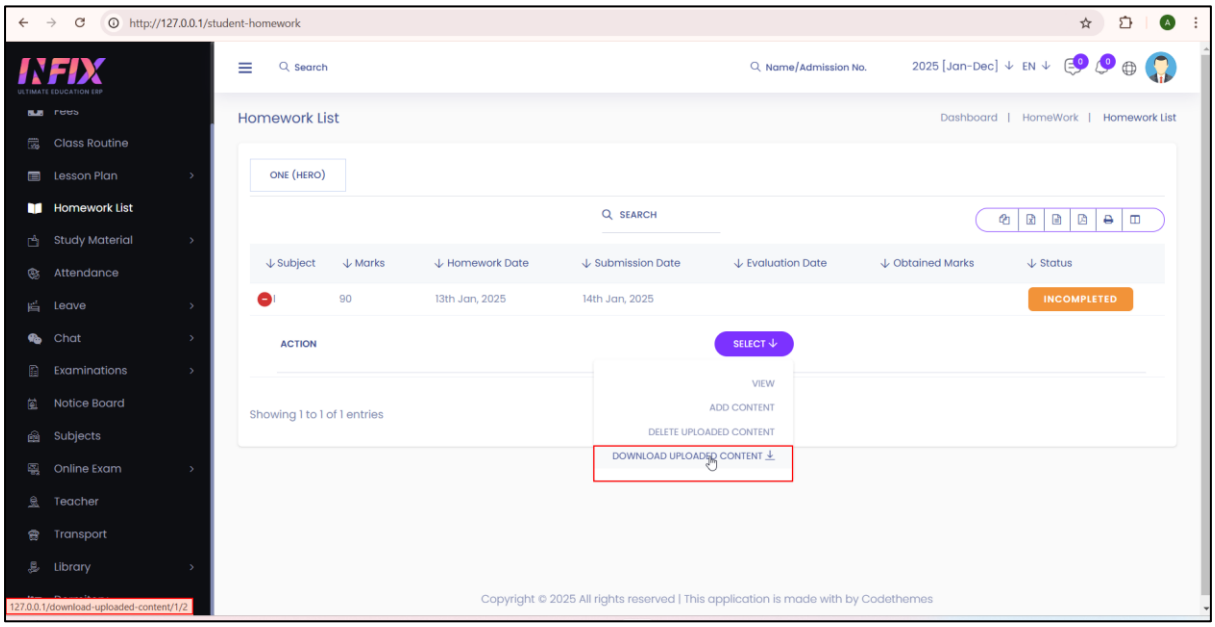
The students of different classes using the application have option to upload their respective HomeWorks. However, it was observed that student and homework have unique ID's which are sequential and guessable. It makes it easy for one student from one class to download homework done by another student of another class.

Steps To Replicate:

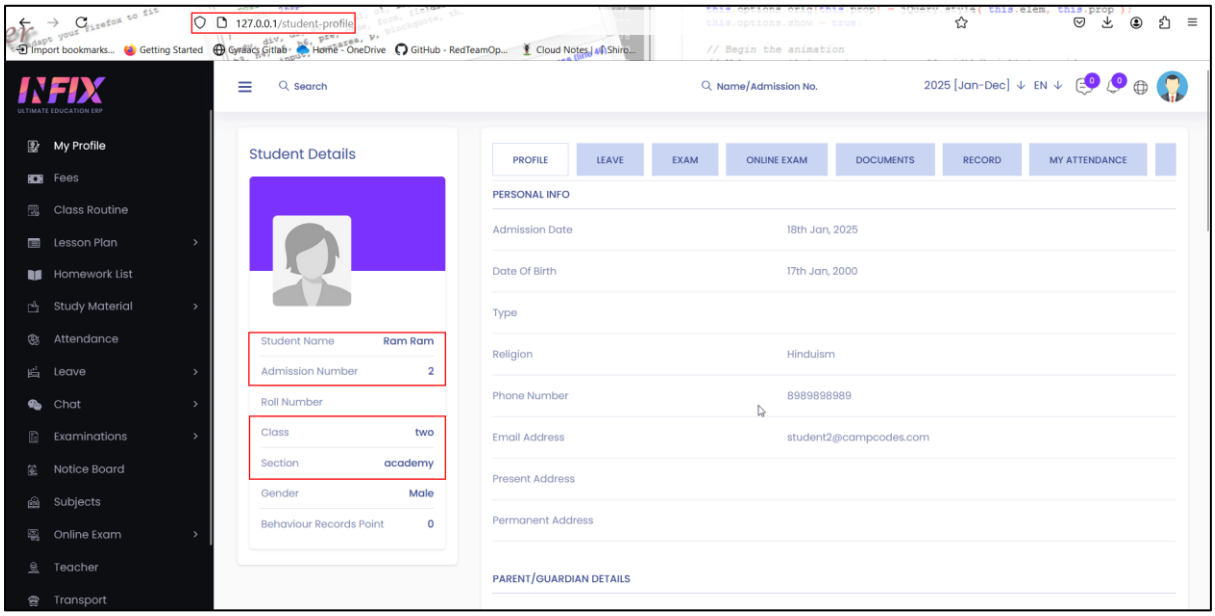
Step 1: Login inside **Student1** user account and traverse to homework section of the application and upload a homework.



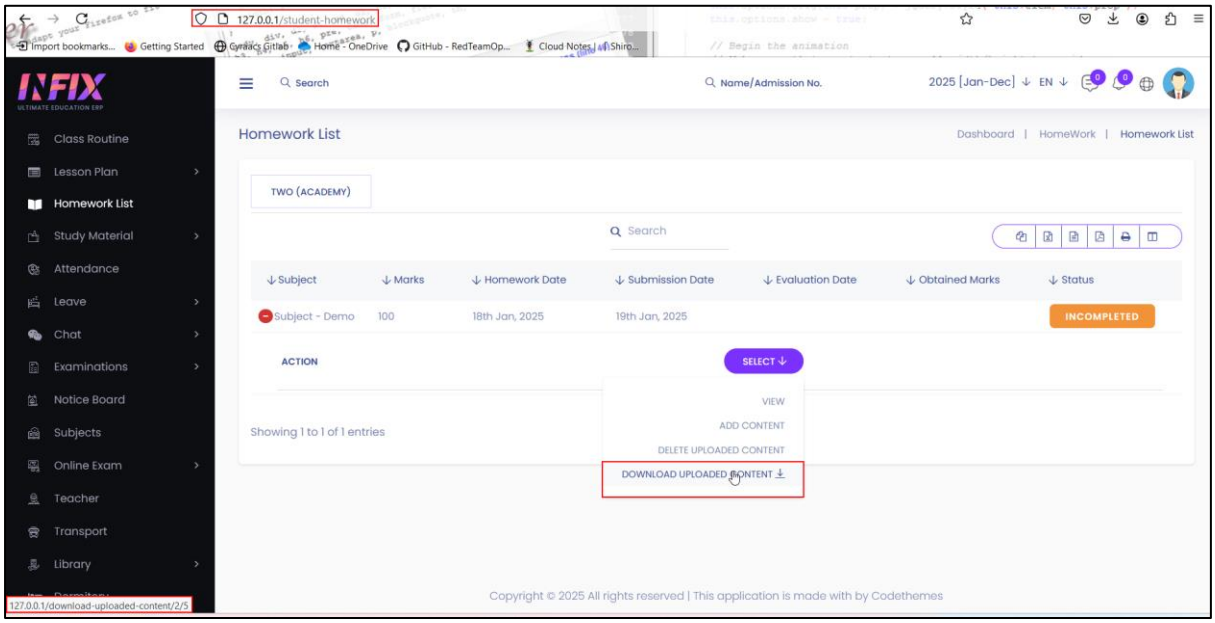
Once homework is uploaded it can be observed a student can download the homework. It can be observed that **Student1** has URL containing **"/1"** indicating **1st Admission** and **"/2"** indicating the attachment number.



Step 2: Similarly, login inside **Student2** user account which is in a completely different class.

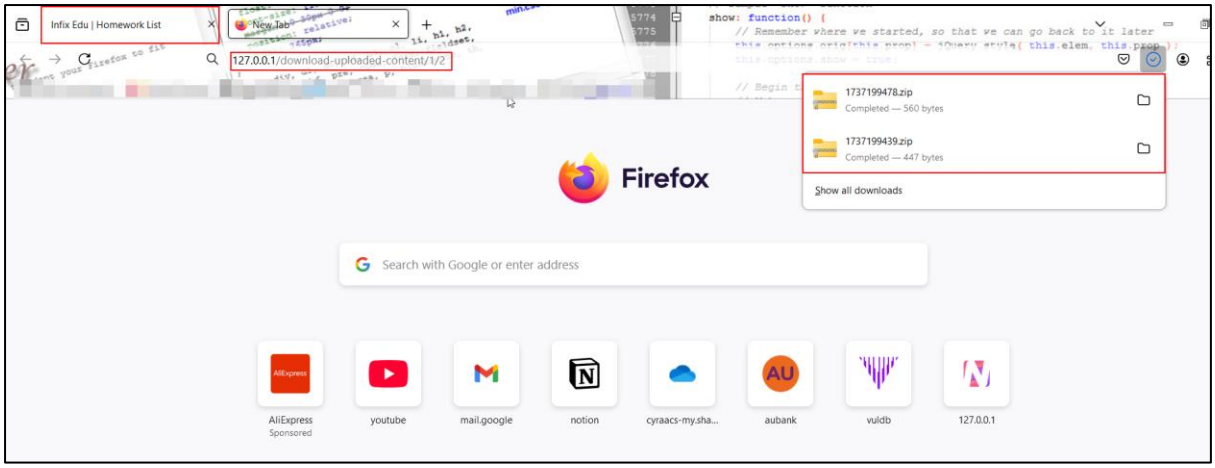


Repeat the same steps of uploading the homework and checking the download URL. It can be observed that **"/2"** in the URL indicates the **2nd Admission Number** and **"/5"** indicating the attached file number.

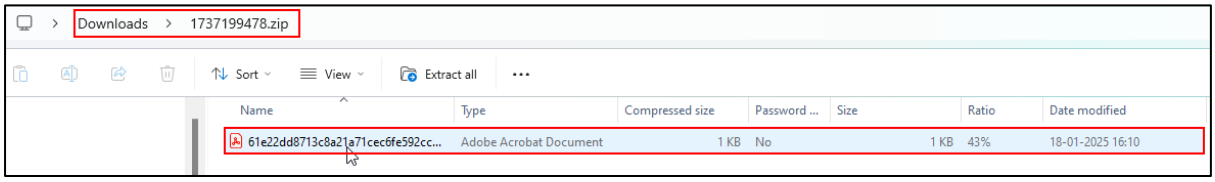


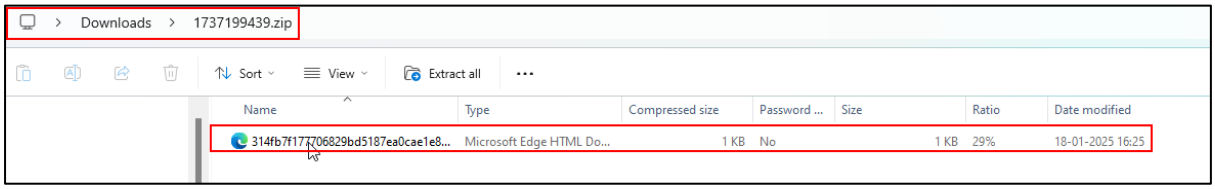
Step 3: From **Student2** Account make a request to download the **5th attachment** which belongs to **Student2** as well as **2nd attachment** which belong to **Student1**.

It can be observed that two different ZIP files could be downloaded. One zip containing homework of **Student2** and another containing homework of **Student1.**



Step 4: Open the Zip files. It can be observed that one zip file has PDF, and another has Image indicating successful IDOR and unauthorized download being allowed to a student to download all other student HomeWorks.





Recommendation:

To mitigate IDOR, implement access control checks for each object that users try to access. Web frameworks often provide ways to facilitate this. Additionally, use complex identifiers as a defence-in-depth measure, but remember that access control is crucial even with these identifiers. Avoid exposing identifiers in URLs and POST bodies if possible. Instead, determine the currently authenticated user from session information. When using multi-step flows, pass identifiers in the session to prevent tampering.