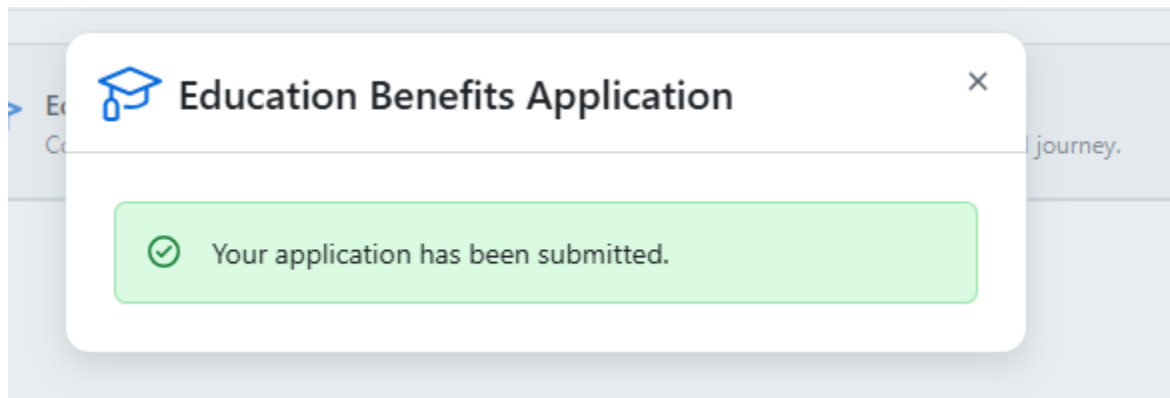Proof of application!

For part c:
Version control tracks all changes to code and timelogs them, allowing teams to work separately on the same project without risking losing past versions/erasing history. Without version control, work could easily get overwritten or lost. Every change is logged with author, date, and notes. This helps a lot with debugging should a new addition screw with the rest of the code. Additionally, it can provide comments to better explain why certain changes were made.

Git is optimized for speed and uses SHA1 hashing to ensure code integrity. It seems quite complex but has a lot of benefits for many different goals.

-

Yes, in past group projects, version control would have been VERY useful. We often overwrote each other's files accidentally when sharing code via email/cloud storage, leading to lost work and confusing "final_v2_updated_updated_final_FINAL(1)(2)" esque file names. This kind of file naming provides no useful metadata nor any explanation by the creator for why a new edition was made. Additionally, it can easily overwrite the correct material, which has caused many many problems before (ie one partner losing the entire project due to laptop issues, the rest of the group having to search through their own files to piece it back together). Using Git would have allowed us to track changes, merge contributions cleanly, and maintain a stable main version while experimenting with new features separately.