För att säkerställa att användarnas lösenord hanteras säkert i min webbapplikation, skulle jag aldrig lagra lösenorden i klartext. Istället skulle jag implementera kryptering genom att använda en stark hash-algoritm med saltning. Jag skulle välja bcrypt som hash-algoritm eftersom den är designad för att vara motståndskraftig mot brute-force-attacker genom att vara beräkningsmässigt krävande. Bcrypt inkluderar också inbyggd saltning, vilket innebär att varje lösenord får ett unikt saltvärde. Saltning fungerar genom att man lägger till en slumpmässig sträng till lösenordet innan det hash:as, vilket gör att även om två användare har samma lösenord kommer deras hashade värden att vara olika. Detta skyddar mot rainbow table-attacker och ökar säkerheten avsevärt.

När det gäller användarrättigheter för databastabellerna i min publika webbapplikation skulle jag definiera specifika behörigheter för olika roller för att skydda data och säkerställa att användare endast har tillgång till relevant information. Till exempel, för tabellen Users skulle jobbsökande och arbetsgivare endast kunna läsa och uppdatera sina egna profiler, medan administratörer har fullständiga rättigheter att skapa, läsa, uppdatera och ta bort användaruppgifter. Allmänheten skulle inte ha någon åtkomst till denna tabell.

För tabellen **JobListings,** skulle arbetsgivare kunna skapa nya jobbannonser samt uppdatera eller ta bort sina egna annonser. Jobbsökande skulle kunna läsa alla jobbannonser för att söka efter lediga tjänster, och administratörer skulle ha fullständiga rättigheter att hantera alla annonser. Allmänheten skulle kunna ha begränsad läsåtkomst för att öka synligheten av annonser.

När det gäller tabellen **Applications,** skulle jobbsökande kunna skapa nya ansökningar och se status på sina egna ansökningar. Arbetsgivare skulle kunna läsa ansökningar som är kopplade till deras egna jobbannonser, och administratörer skulle ha fullständig kontroll för att hantera och moderera ansökningar. Allmänheten skulle inte ha någon åtkomst till denna tabell för att skydda känslig information.

Genom att implementera **bcrypt** för lösenordshantering och noggrant definiera användarrättigheter säkerställer jag att min databas är skyddad och att användarnas integritet bevaras i min webbapplikation. Detta skapar en säker och pålitlig miljö för alla användare och bygger förtroende genom att visa att jag tar deras säkerhet på allvar.