**Databasschema. Tabeller och kolumner i databasen, med tillräckliga förklaringar för att man ska kunna förstå vad innehållet betyder. Ett diagram över schemat i en eller annan form är utmärkt att ha – det gör det lättare att prata om – men inget absolut krav.**

Databasen som applikationen har ett antal tabeller som är nödvändiga för att effektivt kunna lagra data.

Tabellen Köp är viktigt då kunders köp lagras här. De kolumner som finns till denna tabell är Transaktionsid, Kundid, Avgångsid och Antalplatser. Tabellens nycklar är Transaktionsid som primär nyckel samt Kundid (från tabellen Kund) och Avgångsid (från tabellen Resa) som främmande nycklar.

Tabellen Resa är den längsta som finns i databasen då alla destinationer ska lagras här. Kolumnerna Avgångsid, Avgångsland, Avgångsstad, Ankomstland, Ankomststad, Avgångsdatum, Ankomstdatum, Avgångstid, Ankomsttid, Pris, Antalplatser, Chaufförpersonnummer används för att alla resor ska finnas med. Primär nyckel är avgångsid då denna är unik för varje rad. Främmande nycklar är Avgångsland och Avgångsstad (från tabellen Stad) och Chaufförpersonnummer (från tabellen Chaufför).

Tabellen Chaufför används för att hantera chaufförernas personuppgifter så att det sedan går att koppla samman resor och chaufförer. Här sparas chaufförens Personnummer, Förnamn, Efternamn, Adress, Stad och hemtelefon. Den primära nyckeln är chaufförpersonnummer då denna alltid är unik. Tabellen innehåller inga främmande nycklar.

Tabellen Kund är lik Chaufför tabellen då den sparar kunders personuppgifter som Förnamn, Efternamn, Adress, Stad, Hemtelefon och slutligen primärnyckeln Kundid. Denna tabell har inte heller främmande nycklar.

Sista tabellen Stad har endast kolumnerna Land, Stad och Adress. Här sparas alla destinationer separat utan några kopplingar. Tabellen har två primärnycklar Land och Stad och återigen så används inga främmande nycklar.

**Redogörelse vad ni använt för att bygga applikationen. Programspråk, kodbibliotek osv.**

I enlighet med kursens krav används en databas, språket för databasen i fråga är postgresql. Gruppen har utvecklat en webbaserad applikation för Mörtfors buss. Denna webbapplikation bestående av html, css och javascript är baserad på ramverket bootstrap.

För att koppla ihop webbapplikationen med postgresql så har flask som är ett ramverk för python använts. För att brygga ihop python och postgresql så har databasadaptern psycopg använts.

**Kortfattad överblick över applikationens komponenter, från programmerarens perspektiv, och beskrivning vad de gör.**

Applikationen använder Python 3.6 för backend. Python används i kombination med Flask som är ett ramverk vars användningsområde är att utveckla hemsidor och applikationer. Flask och Python kommunicerar med databasen vilket innebär att alla sql-satser skrivs här. Psycopg är även ett bibliotek som inkluderats då detta används för att snabbt och lätt skapa en anslutning till databasen. När detta är gjort så förs informationen över till html för att presenteras för användaren. Html dokumenten används för att visa och strukturera informationen, bootstrap används i kombination för att ge webbsidan style samt att hjälpa till med modal fönster.

**Kortfattad beskrivning av hur man använder er applikation (eller era applikationer) från användarperspektiv.**

Då applikationens primära mål är för gruppen att uppvisa färdighet i postgresql så har valet gjorts att applikationen är utformad att kunna användas av både Mörtfors buss och resenärer under samma gränssnitt. Hade kursens krav varit att utveckla en applikation för ett riktigt företag så hade applikationen delats upp i två applikationer. Ett program för Mörtfors buss samt ett program för resenärer.

Applikationen innefattar flertalet olika delar som är sammankopplade och i vissa fall beroende av varandra. I stora drag är applikationen utformad så att en användare kan registrera kunder, registrera chaufförer, registrera hållplatser, och avgångar i respektive register. Det finns även möjlighet att se transaktioner dvs kunder som köpt resor.

Utöver denna funktion finns även möjlighet att söka efter avgångar samt köpa biljetter för respektive avgångar. Sökfunktionen möjliggör att specificera sin sökning så som: resor från Malmö eller resor som körs av chaufför x.

Då olika tabeller har sammanband och dessa sammanband är primärnycklar så är också de olika delarna beroende av varandra. Det går tex inte att skapa en avgång från London om inte först London finns registrerat i hållplatser. Om en kund försöker köpa fler biljetter än vad som finns kvar för en avgång går inte köpet igenom.

Applikationen ger även feedback till användaren i form av gröna eller röda textfält efter att användaren utfört något.