Technisch Ontwerp

**Toelichting gekozen oplossingen:**

**Memory:**

Bij het memory spel hebben we een ‘Main’ bestand waar we functies oproepen van ‘Scherm’ en ‘Logica’. We hebben er voor gekozen om de foto’s in een arraylist te doen in paren zodat het gemakkelijk aan de buttons toegevoegd kan worden. Als je twee dezelfde foto’s vindt, blijven ze staan en trekken we 2 van de aantal buttons variabele af. Als het aantal buttons variabele 0 is, heb je gewonnen.

**MySql:**

Bij de MySql opdracht hebben we er voor gekozen om aan het begin een connectie te maken met de database en dan alles te pakken van de database. Als je op een knop klikt gaat de value van de invoerbalk in de database en wordt het in een panel op het scherm gezet.

**Eindproducten:**

**Memory:**

Bestaat uit: een klasse met getter functies en wat functies om de variabelen zoals rijen en kolommen te setten en een arraylist met foto’s,

een klasse met functies die buttons op het scherm laat verschijnen en

een klasse waar alles uitgevoerd wordt oftewel de ‘Main.java’ bestand.

**MySql:**

Bestaat uit:

een databaseserver genaamd: personen.sql,

een klasse met functies die: de connectie start, alle informatie pakt van de database en die informatie in de database doet en een klasse waar buttons en textfields op het scherm gezet worden.

**Kwaliteiteisen:**

* Er is 1 client nodig.
* De database heet ‘personen’, de username is ‘jeffrey’ en het wachtwoord is ‘Anglito22@’.
* 1 persoon kan memory gebruiken en iedereen kan mysql gebruiken.
* Er gelden geen afspraken.
* Bij memory worden de foto’s in een folder gedaan.

**Afhankelijkheden:**

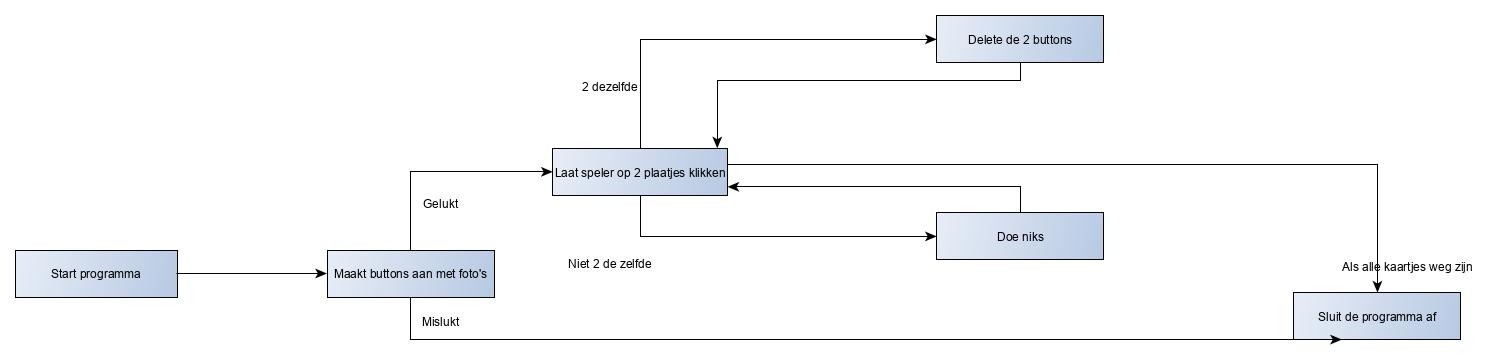
**Memory:**

De ‘Main’ bestand heft de andere bestanden zoals ‘Logica.java’ en ‘Scherm.java’ nodig omdat het functies oproept van die klasses.

**MySql:**

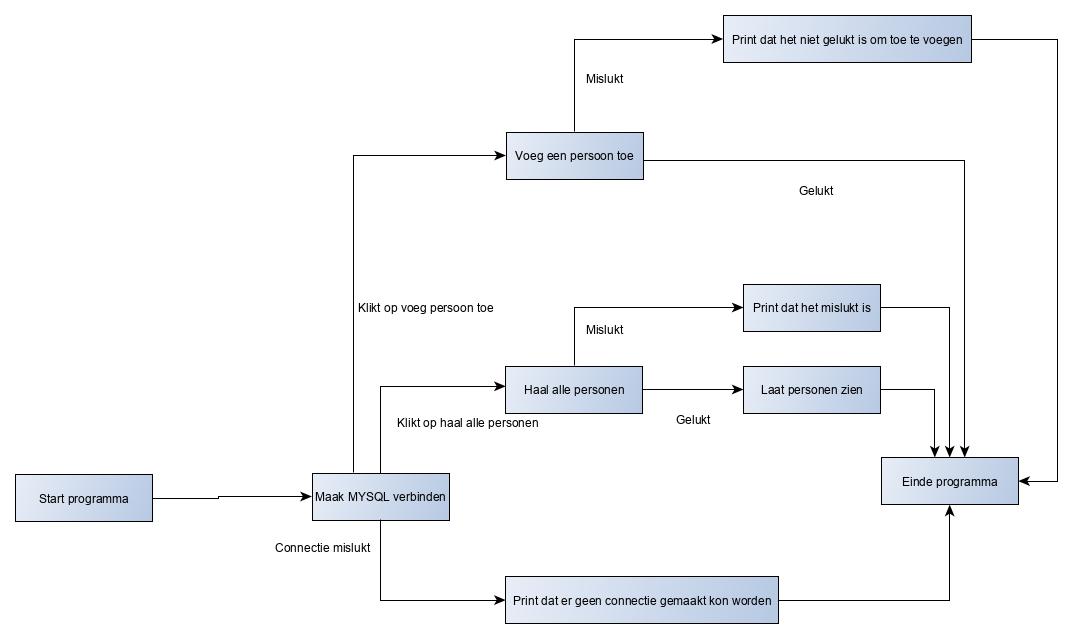
Je hebt de ‘personen.sql’ bestand nodig om er informatie in te doen en van te pakken.

‘Programma.java’ heeft ‘Database.java’ nodig om om functies op te roepen.



**Structuren:**

**Memory:**



**MySql:**