COMP3111 Project
Group: T-28

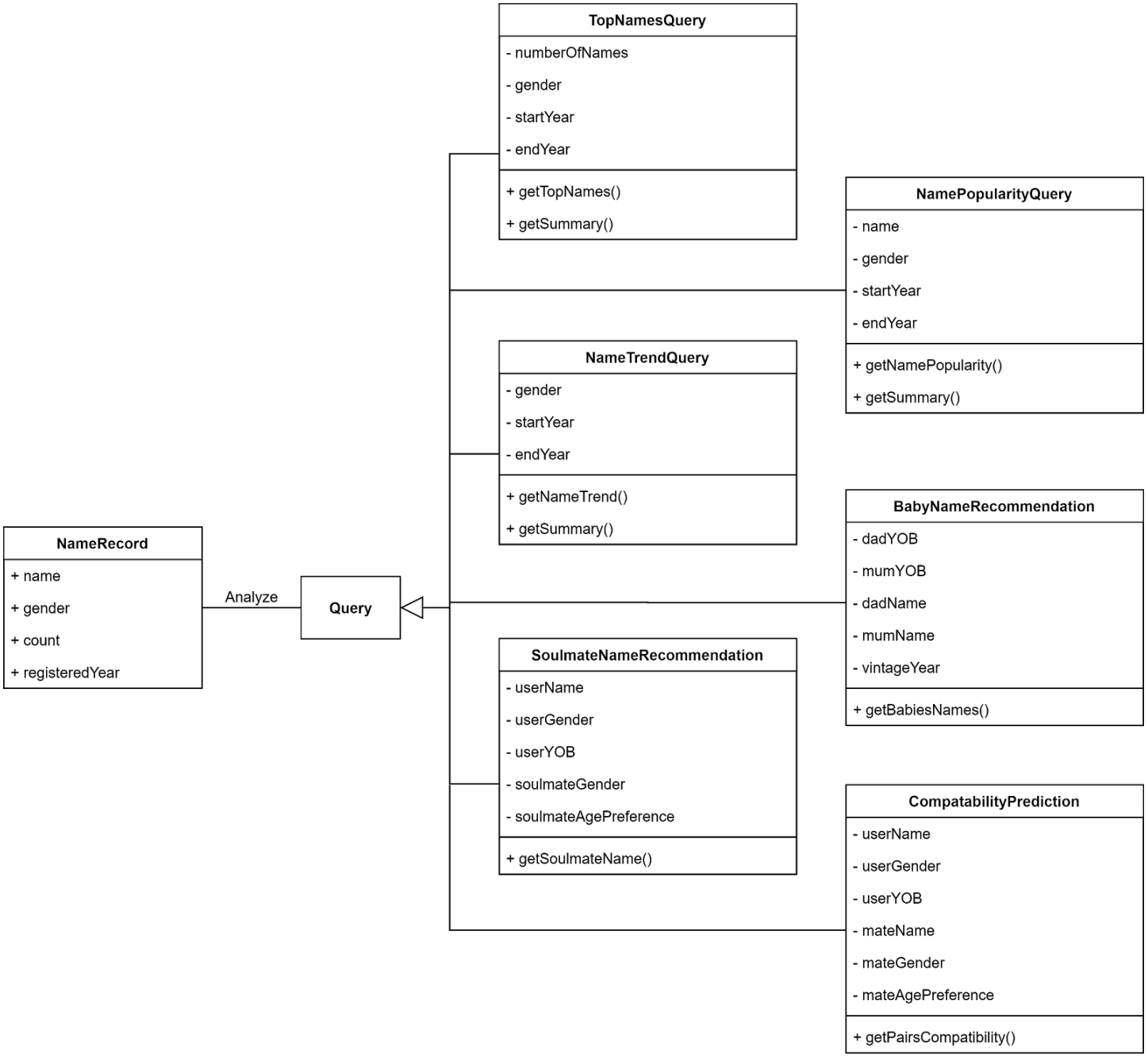
1. PANG, Kit (kpangaa)
Task 3 & Task 6
Representing Background Color: Blue

2. TAM, Tsz Chung (tctam)
Task 2 & Task 5
Representing Background Color: Green

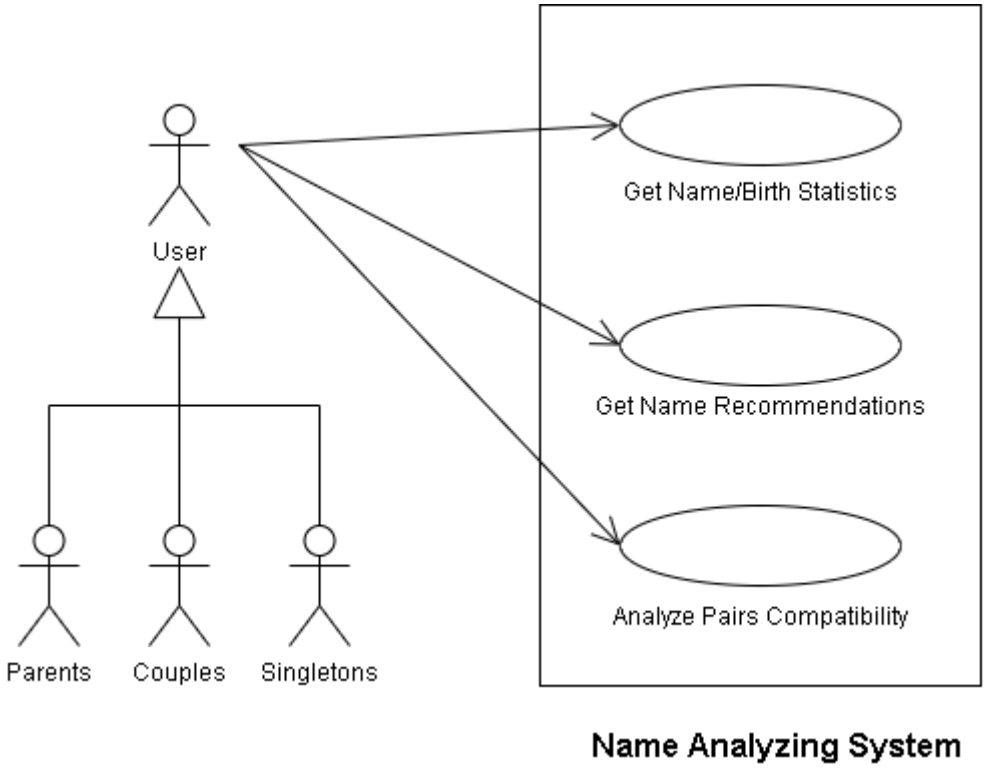
3. WU, Tsz Yeung (tywuab)
Task 1 & Task 4
Representing Background Color: Yellow

(Note: Parts without background color indicate two or more students collaborated)

Class Diagram



Use-case Diagram



Note:

1. **Task 0, 1, 2, 3 are combined to one use case "Get Name/Birth Statistics"**
   Reason:
   These tasks have similar nature in that they all query statistics of name or birth.

2. **Task 4 and 5 are combined to one use case "Get Name Recommendations"**
   Reason:
   These two tasks have similar nature in that both of them generate names from the data set based on an algorithm.

3. **Task 6 is one use case "Analyze Pairs Compatibility" by itself**
   Reason:
   It does not share a similar nature with any other task and it is not a partial functionality of any other use case.

Use-case Specification

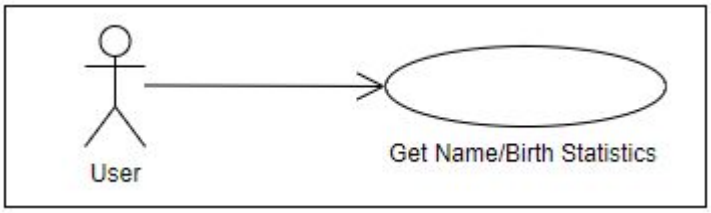
Task 1, 2, 3

Use Case: Get Birth/Name Statistics

Brief Description

The use case describes how a user queries the statistics on birth and name, including top names at birth, the popularity of the name, and the trend of the popularity of the name.

Use-case Diagram



Basic Flow

1. The use case begins when the user actor chooses to query particular statistics on names and births.
2. While the user has a query to perform
   - 2.1. If the user chooses to query the top names for birth by selecting the tab "Reporting 1"
     - 2.1.1. The system displays the interface for querying the top names at birth.

     {Enter Data For Top Names For Birth}
     - 2.1.2. The user inputs "n", to get the top n most frequently registered names, the gender of the names, and the period of interest, then starts the query.

     {Start Query For Top Names For Birth}
     - 2.1.3. The system retrieves data on the top "n" names for birth from the data set according to the inputted parameters
     - 2.1.4. The system displays a detailed result of the query illustrated with a table and provides a summary of the results.
   - 2.2. If the user chooses to query the popularity of a name over a particular period by selecting the tab "Reporting 2"
     - 2.2.1. The system displays the interface for querying the popularity of a particular name over a particular period.

     {Enter Data For Popularity Query}
     - 2.2.2. The user inputs the name to be queried, gender for the name, and the period of interest(indicated by 2 inputted years), then starts the query.

     {Begin Popularity Query}
     - 2.2.3. The system retrieves data on the popularity of the name over the period from the data set.
     - 2.2.4. The system displays the result with a table, and provides a corresponding summary.

2.3. If the user chooses to query the trend in popularity of names over a particular period by selecting tab “Reporting 3”

2.3.1. The system displays the interface for querying the trend of names over a particular period.

{Enter Data For Trend Query}

2.3.2. The user indicates the gender and the period (indicated by 2 different years) to be queried, then starts the query.

{Begin Trend Query}

2.3.3. The system retrieves data on the trend of names over the period from the data set.

2.3.4. The system displays the result with a table and provides a corresponding summary.

3. The use case ends.

Task 1, 2, 3

Alternative Flows

A1: Invalid Data For Query For Top Names For Birth

At {Start Query For Top Names For Birth} if any of the data entered is invalid,

1. The system informs the user that the data entered is invalid.
2. The flow of events is resumed at {Enter Data For Top Names For Birth}.

A2: Invalid Inputs For Popularity Query

At {Begin Popularity Query} if any of the inputted data is invalid,

1. The system informs the user that the input data is invalid.
2. The flow of events is resumed at {Enter Data For Popularity Query}.

A3: No Data For Popularity Of Queried Name

At {Begin Popularity Query} if the queried name is not ranked in the whole period,

1. The system informs the user that the queried name is not ranked in the whole period.
2. The flow of events is resumed at {Enter Data For Popularity Query}.

A4: Invalid Inputs For Trend Query

At {Begin Trend Query} if the inputs are invalid,

1. The system informs the user that the inputs are invalid.
2. The flow of events is resumed at {Enter Data For Trend Query}.

A5: No Trend For Queried Period

At {Begin Trend Query} if no computable trend in the data set for the queried period,

1. The system informs the user that there is no trend for the queried period.
2. The flow of events is resumed at {Enter Data For Trend Query}.

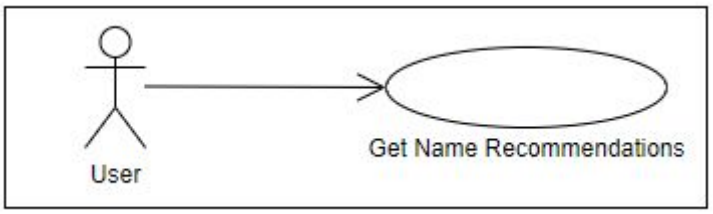
Task 4, 5

Use Case: Get Name Recommendations

Brief Description

The use case describes how a user gets recommendations on names for newborn babies or recommendations on names for his/her compatible soulmate.

Use-case Diagram



Basic Flow

1. The use case begins when the user chooses to get recommendations on names for newborn babies or for a possible soulmate.
2. While the user wants to get recommendation on names
   - 2.1. If the user chooses to get recommendation on names for newborn babies by selecting tab “Application 1”
     - 2.1.1. The system displays the interface for getting the recommended name for newborn babies.

     {Enter Data For Baby Name Recommendation}
     - 2.1.2. The user will enter the name and year of birth of both parents, and an optional year, then starts the query.

     {Start Query For Baby Name Recommendation}
     - 2.1.3. The system computes the recommended name for a baby according to the input.

     {Display Recommended Name For Babies}
     - 2.1.4. The system displays the recommended name for a baby according to the gender specified.
   - 2.2. If the user chooses to get recommendations (i.e. predictions) on the name for a compatible soulmate by selecting tab “Application 2”
     - 2.2.1. The system displays the interface for getting the recommended name for a compatible soulmate.

     {Enter Data For Soulmate Name Recommendation}
     - 2.2.2. The user indicates his/her name, gender, year of birth, the preferred gender of the soulmate to be matched, and the preference on having a younger or older soulmate, then starts the query.

     {Begin Soulmate Name Recommendation}
     - 2.2.3. The system computes the recommended name of the soulmate according to the inputs.

     {Display Recommended Name For Soulmate}
     - 2.2.4. The system displays the recommended name of the soulmate.
3. The use case ends.

Task 4, 5

Alternative Flows

A1: Invalid Input Data For Baby Name Recommendation

At {Start Query For Baby Name Recommendation} if any of the data inputted is invalid,

1. The system will use default data in place of the invalid data
2. The flow of events is resumed at {Enter Data For Baby Name Recommendation}

A2: No Available Names Computed

At {Start Query For Baby Name Recommendation} if there is no available name at that computed rank,

1. The system will notify the user that a name at that rank is not found and will recommend the top name of that year to the user.
2. The system will return the top-ranked name of that year.
3. The flow of events is resumed at {Display Recommended Name For Babies}.

A3: Invalid Inputs For Soulmate Name Recommendation

At {Begin Soulmate Name Recommendation} if any of the inputted data is invalid,

1. The system informs the user that the input data is invalid.
2. The flow of events is resumed at {Enter Data For Soulmate Name Recommendation}.

A4: No Reasonable Soulmate Name Recommendation

At {Begin Soulmate Name Recommendation} if there is no name satisfying the specified soulmate's gender, computed rank and year of birth,

1. The system will inform the user that no reasonable recommendation can be made and will recommend the top-ranked name.
2. The system will return the top-ranked name of the specified gender in the computed soulmate's year of birth.
3. The flow of events is resumed at {Display Recommended Name For Soulmate}.

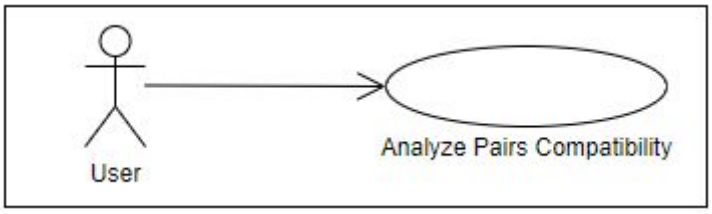
Task 6

Use Case: Analyze Pairs Compatibility

Brief Description

This use case describes how a user queries the score prediction of the compatibility between the user and the person to be matched.

Use-case Diagram



Basic Flow

1. The use case begins when the user chooses to query the score prediction of the compatibility between the user and the person to be matched, by selecting tab “Application 3”.
2. The system displays the interface of querying the score prediction.
3. While the user has a query to perform

   {Enter Data}

   3.1. The user indicates the name, gender and year of birth of himself/herself, name and gender of the person to be matched, and his/her preference on the person to be matched being younger or older than him, then starts the query.

   {Begin Query}

   3.2. The system computes the score of the compatibility with the given personal information and preference.

   {Display Result}

   3.3. The system displays the score of compatibility.
4. The use case ends.

Alternative Flows

A1: Invalid Input Data

At {Begin Query} if any of the inputted data is invalid,

1. The system informs the user that the input data is invalid.
2. The flow of events is resumed at {Enter Data}.

A2: No Rank For Queried Data

At {Begin Query} if any of the queried names is not ranked in the corresponding queried gender in the queried year of birth,

1. The system informs the user that the queried data is unranked.
2. The system substitutes the corresponding empty rank with the default value.
3. The system computes the score of the compatibility using the substituted value.
4. The flow of events is resumed at {Display Result}.