section *homestay* **parents** *standard_toolkit*

Contents

1 General

section *general* **parents** *homestay*, *update*

We want to have a few basic types here. We need to know general information about all of the users in the system:

- Email address, this is the unique identifier in the system
- Password, this has restrictions:
  - At least 8 characters in length
  - 1 digit
  - 1 uppercase character
  - 1 lowercase character
- First name, should be non-empty
- Last name, should be non-empty

$[Email, Password, FirstName, LastName]$

$$
\begin{aligned}
&Month ::= January\\
&\quad \mid February\\
&\quad \mid March\\
&\quad \mid April\\
&\quad \mid May\\
&\quad \mid June\\
&\quad \mid July\\
&\quad \mid August\\
&\quad \mid September\\
&\quad \mid October\\
&\quad \mid November\\
&\quad \mid December\\
&Day == \mathbb{N}\\
&Year == \mathbb{N}
\end{aligned}
$$

We want to know whether or not the user is an admin or not. This is used so that they get access to certain features in the application.

$$
\begin{aligned}
&AdminFlag ::=\\
&\quad Admin\\
&\quad \mid NotAdmin
\end{aligned}
$$

$$
\begin{aligned}
&\mathit{InappropriateFlag} ::= \\
&\quad \mathit{Zero} \\
&\quad \mid \mathit{First} \\
&\quad \mid \mathit{Second} \\
&\quad \mid \mathit{Third} \\
&\quad \mid \mathit{Remove}
\end{aligned}
$$

Using these basic types we construct more complex types in the system. The applicant is the user of the system. Password tokens are used to reset the password.

$$
\begin{aligned}
&\mathit{Applicant} == \mathit{Email} \times \mathit{Password} \times \mathit{FirstName} \times \mathit{LastName} \times \mathit{AdminFlag} \\
&PToken == \mathbb{N} \times \mathit{Email} \\
&\mathit{Date} == \mathit{Month} \times \mathit{Day} \times \mathit{Year} \\
&\mathit{IFlag} == \mathit{Email} \times \mathit{InappropriateFlag}
\end{aligned}
$$

We need to be able to notify the user of what is happening in the application. So, we enumerate the possible responses from actions taken.

$$
\begin{aligned}
&\mathit{Response} ::= \\
&\quad InvalidToken \\
&\quad \mid PasswordResetSuccessful \\
&\quad \mid \mathit{LoginSuccessful} \\
&\quad \mid InvalidAvailability \\
&\quad \mid ValidAvailability
\end{aligned}
$$

We need to know about the host stuff.

$$
\begin{array}{l}
\mathit{Smoking} ::= \\
\quad \mathit{EnjoysSmoking} \\
\quad \mid \mathit{NonSmoking} \\
\mathit{Pets} ::= \\
\quad \mathit{NoPets} \\
\quad \mid \mathit{YesPets} \\
\mathit{Children} ::= \\
\quad \mathit{One} \\
\quad \mid \mathit{Two} \\
\quad \mid \mathit{ThreePlus} \\
\mathit{Diet} ::= \\
\quad \mathit{GlutenFree} \\
\quad \mid \mathit{Omnivore} \\
\quad \mid \mathit{Pescatarian} \\
\quad \mid \mathit{Vegetarian} \\
\quad \mid \mathit{Vegan} \\
\quad \mid \mathit{OtherDiet} \\
\mathit{Religion} ::= \\
\quad \mathit{Agnostic} \\
\quad \mid \mathit{Athiest} \\
\quad \mid \mathit{Buddhist} \\
\quad \mid \mathit{Christian} \\
\quad \mid \mathit{Catholic} \\
\quad \mid \mathit{Mormon} \\
\quad \mid \mathit{Muslim} \\
\quad \mid \mathit{OtherReligion}
\end{array}
$$

Does the host want to post the exact location of his/her home or a general area where the home exists inside of?

$$\mathit{ExactOrArea} ::= \mathit{ExactLocation} \mid \mathit{Area}$$

$$[\mathit{StreetNumber}, \mathit{StreetName}, \mathit{City}, \mathit{State}, \mathit{ZipCode}, \mathit{Phone}, \mathit{Photo}]$$

$$
\begin{array}{l}
\mathit{PostingPhoto} == \mathit{Photo} \\
\mathit{UserPhoto} == \mathit{Photo}
\end{array}
$$

$Price == \mathbb{N}$
$SizeOfRoom == \mathbb{N} \times \mathbb{N}$
$Address == StreetNumber \times StreetName \times City \times State \times ZipCode$
$AddressInfo == Address \times ExactOrArea$
$Availability == Date \times Date$
$HostPreference == Applicant \times Smoking \times Pets \times Children \times Diet \times Religion$
$Posting == HostPreference \times AddressInfo \times SizeOfRoom \times Price \times Availability \times PostingPhoto$

Our initial state is just a bunch of empty sets.

$$
\begin{array}{|l}
\hline
\textit{HomestayInitial} \\
Applicants : \mathbb{P}\, Applicant \\
Emails : \mathbb{P}\, Email \\
Passwords : \mathbb{P}\, Password \\
PhoneNumbers : \mathbb{P}\, Phone \\
UserPhoto : \mathbb{P}\, Photo \\
Valid, Invalid : \mathbb{P}\, PToken \\
IFlags : \mathbb{P}\, IFlag \\
Postings : \mathbb{P}\, Posting \\
HostPreferences : \mathbb{P}\, HostPreference \\
\hline
Applicants = \varnothing \\
Emails = \varnothing \\
Passwords = \varnothing \\
PhoneNumbers = \varnothing \\
UserPhoto = \varnothing \\
Valid = Invalid = \varnothing \\
IFlags = \varnothing \\
Postings = \varnothing \\
HostPreferences = \varnothing \\
\hline
\end{array}
$$

In the database we need to ensure that there is each email address is unique, and that the password tokens are either valid or invalid.

$$
\begin{array}{|l}
\hline
\textit{HomestayDatabase} \\
HomestayInitial \\
\hline
\forall\, e_0, e_1 : Email;\ p_0, p_1 : Password;\ fn_0, fn_1 : FirstName;\ ln_0, ln_1 : LastName \bullet \\
\qquad e_0 = e_1 \Rightarrow (p_0 = p_1 \land fn_0 = fn_1 \land ln_0 = ln_1) \\
Valid \cap Invalid = \varnothing \\
\hline
\end{array}
$$

1.1 General Operators

It gets kind of silly to have to rewrite these operators and functions each time, so we have some helpers here.

We want an easy way to update an applicant. This helper function updates the applicant in the set of all applicants, identified by the email address.

$$
\begin{array}{l}
[A, B, C, D, E, F] \\
\hline
\mathit{firstOf}3 == \lambda\, a : A;\; b : B;\; c : C \bullet a \\
\mathit{secondOf}3 == \lambda\, a : A;\; b : B;\; c : C \bullet b \\
\mathit{thirdOf}3 == \lambda\, a : A;\; b : B;\; c : C \bullet c \\
\mathit{firstOf}4 == \lambda\, a : A;\; b : B;\; c : C;\; d : D \bullet a \\
\mathit{secondOf}4 == \lambda\, a : A;\; b : B;\; c : C;\; d : D \bullet b \\
\mathit{thirdOf}4 == \lambda\, a : A;\; b : B;\; c : C;\; d : D \bullet c \\
\mathit{fourthOf}4 == \lambda\, a : A;\; b : B;\; c : C;\; d : D \bullet d \\
\mathit{firstOf}5 == \lambda\, a : A;\; b : B;\; c : C;\; d : D;\; e : E \bullet a \\
\mathit{secondOf}5 == \lambda\, a : A;\; b : B;\; c : C;\; d : D;\; e : E \bullet b \\
\mathit{thirdOf}5 == \lambda\, a : A;\; b : B;\; c : C;\; d : D;\; e : E \bullet c \\
\mathit{fourthOf}5 == \lambda\, a : A;\; b : B;\; c : C;\; d : D;\; e : E \bullet d \\
\mathit{fifthOf}5 == \lambda\, a : A;\; b : B;\; c : C;\; d : D;\; e : E \bullet e \\
\mathit{firstOf}6 == \lambda\, a : A;\; b : B;\; c : C;\; d : D;\; e : E;\; f : F \bullet a \\
\mathit{secondOf}6 == \lambda\, a : A;\; b : B;\; c : C;\; d : D;\; e : E;\; f : F \bullet b \\
\mathit{thirdOf}6 == \lambda\, a : A;\; b : B;\; c : C;\; d : D;\; e : E;\; f : F \bullet c \\
\mathit{fourthOf}6 == \lambda\, a : A;\; b : B;\; c : C;\; d : D;\; e : E;\; f : F \bullet d \\
\mathit{fifthOf}6 == \lambda\, a : A;\; b : B;\; c : C;\; d : D;\; e : E;\; f : F \bullet e \\
\mathit{sixthOf}6 == \lambda\, a : A;\; b : B;\; c : C;\; d : D;\; e : E;\; f : F \bullet f
\end{array}
$$

1.2 General Updates

section *general_updates* **parents** *homestay*

Updating for 3-tuples.

$$
\begin{array}{l}
[A, B, C] \\
\hline
\mathit{updateFirstOf}3 : (A \times B \times C) \times A \rightarrow \\
\qquad\qquad A \times B \times C \\
\hline
\forall\, abc : A \times B \times C;\; a : A \bullet \\
\qquad \exists\, a_0 : A;\; b : B;\; c : C \mid abc = (a_0, b, c) \bullet \\
\qquad\qquad \mathit{updateFirstOf}3(abc, a) = (a, b, c)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C] \\
\hline
\mathit{updateSecondOf3} : (A \times B \times C) \times B \rightarrow \\
\qquad\qquad\qquad A \times B \times C \\
\hline
\forall abc : A \times B \times C;\ b : B \bullet \\
\qquad \exists a : A;\ b_0 : B;\ c : C \mid abc = (a, b_0, c) \bullet \\
\qquad\qquad \mathit{updateSecondOf3}(abc, b) = (a, b, c)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C] \\
\hline
\mathit{updateThirdOf3} : (A \times B \times C) \times C \rightarrow \\
\qquad\qquad\qquad A \times B \times C \\
\hline
\forall abc : A \times B \times C;\ c : C \bullet \\
\qquad \exists a : A;\ b : B;\ c_0 : C \mid abc = (a, b, c_0) \bullet \\
\qquad\qquad \mathit{updateThirdOf3}(abc, c) = (a, b, c)
\end{array}
$$

Updating for 4-tuples.

$$
\begin{array}{l}
[A, B, C, D] \\
\hline
\mathit{updateFirstOf4} : (A \times B \times C \times D) \times A \rightarrow \\
\qquad\qquad\qquad A \times B \times C \times D \\
\hline
\forall abcd : A \times B \times C \times D;\ a : A \bullet \\
\qquad \exists a_0 : A;\ b : B;\ c : C;\ d : D \mid abcd = (a_0, b, c, d) \bullet \\
\qquad\qquad \mathit{updateFirstOf4}(abcd, a) = (a, b, c, d)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D] \\
\hline
\mathit{updateSecondOf4} : (A \times B \times C \times D) \times B \rightarrow \\
\qquad\qquad\qquad A \times B \times C \times D \\
\hline
\forall abcd : A \times B \times C \times D;\ b : B \bullet \\
\qquad \exists a : A;\ b_0 : B;\ c : C;\ d : D \mid abcd = (a, b_0, c, d) \bullet \\
\qquad\qquad \mathit{updateSecondOf4}(abcd, b) = (a, b, c, d)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D] \\
\hline
\mathit{updateThirdOf4} : (A \times B \times C \times D) \times C \rightarrow \\
\qquad\qquad\qquad A \times B \times C \times D \\
\hline
\forall abcd : A \times B \times C \times D;\ c : C \bullet \\
\qquad \exists a : A;\ b : B;\ c_0 : C;\ d : D \mid abcd = (a, b, c_0, d) \bullet \\
\qquad\qquad \mathit{updateThirdOf4}(abcd, c) = (a, b, c, d)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D] \\
\hline
\mathit{updateFourthOf}4 : (A \times B \times C \times D) \times D \rightarrow \\
\qquad\qquad A \times B \times C \times D \\
\hline
\forall\, \mathit{abcd} : A \times B \times C \times D;\ d : D \bullet \\
\quad \exists\, a : A;\ b : B;\ c : C;\ d_0 : D \mid \mathit{abcd} = (a, b, c, d_0) \bullet \\
\qquad \mathit{updateFourthOf}4(\mathit{abcd}, d) = (a, b, c, d)
\end{array}
$$

Updating for 5-tuples.

$$
\begin{array}{l}
[A, B, C, D, E] \\
\hline
\mathit{updateFirstOf}5 : (A \times B \times C \times D \times E) \times A \rightarrow \\
\qquad\qquad A \times B \times C \times D \times E \\
\hline
\forall\, \mathit{abcde} : A \times B \times C \times D \times E;\ a : A \bullet \\
\quad \exists\, a_0 : A;\ b : B;\ c : C;\ d : D;\ e : E \mid \mathit{abcde} = (a_0, b, c, d, e) \bullet \\
\qquad \mathit{updateFirstOf}5(\mathit{abcde}, a) = (a, b, c, d, e)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D, E] \\
\hline
\mathit{updateSecondOf}5 : (A \times B \times C \times D \times E) \times B \rightarrow \\
\qquad\qquad A \times B \times C \times D \times E \\
\hline
\forall\, \mathit{abcde} : A \times B \times C \times D \times E;\ b : B \bullet \\
\quad \exists\, a : A;\ b_0 : B;\ c : C;\ d : D;\ e : E \mid \mathit{abcde} = (a, b_0, c, d, e) \bullet \\
\qquad \mathit{updateSecondOf}5(\mathit{abcde}, b) = (a, b, c, d, e)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D, E] \\
\hline
\mathit{updateThirdOf}5 : (A \times B \times C \times D \times E) \times C \rightarrow \\
\qquad\qquad A \times B \times C \times D \times E \\
\hline
\forall\, \mathit{abcde} : A \times B \times C \times D \times E;\ c : C \bullet \\
\quad \exists\, a : A;\ b : B;\ c_0 : C;\ d : D;\ e : E \mid \mathit{abcde} = (a, b, c_0, d, e) \bullet \\
\qquad \mathit{updateThirdOf}5(\mathit{abcde}, c) = (a, b, c, d, e)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D, E] \\
\hline
\mathit{updateFourthOf}5 : (A \times B \times C \times D \times E) \times D \rightarrow \\
\qquad\qquad A \times B \times C \times D \times E \\
\hline
\forall\, \mathit{abcde} : A \times B \times C \times D \times E;\ d : D \bullet \\
\quad \exists\, a : A;\ b : B;\ c : C;\ d_0 : D;\ e : E \mid \mathit{abcde} = (a, b, c, d_0, e) \bullet \\
\qquad \mathit{updateFourthOf}5(\mathit{abcde}, d) = (a, b, c, d, e)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D, E] \\
\hline
\mathit{updateFifthOf}5 : (A \times B \times C \times D \times E) \times E \rightarrow \\
\qquad\qquad\qquad A \times B \times C \times D \times E \\
\hline
\forall\, \mathit{abcde} : A \times B \times C \times D \times E;\ e : E \bullet \\
\quad \exists\, a : A;\ b : B;\ c : C;\ d : D;\ e_0 : E \mid \mathit{abcde} = (a, b, c, d, e_0) \bullet \\
\qquad \mathit{updateFifthOf}5(\mathit{abcde}, e) = (a, b, c, d, e)
\end{array}
$$

Updating for 6-tuples.

$$
\begin{array}{l}
[A, B, C, D, E, F] \\
\hline
\mathit{updateFirstOf}6 : (A \times B \times C \times D \times E \times F) \times A \rightarrow \\
\qquad\qquad\qquad A \times B \times C \times D \times E \times F \\
\hline
\forall\, \mathit{abcdef} : A \times B \times C \times D \times E \times F;\ a : A \bullet \\
\quad \exists\, a_0 : A;\ b : B;\ c : C;\ d : D;\ e : E;\ f : F \mid \mathit{abcdef} = (a_0, b, c, d, e, f) \bullet \\
\qquad \mathit{updateFirstOf}6(\mathit{abcdef}, a) = (a, b, c, d, e, f)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D, E, F] \\
\hline
\mathit{updateSecondOf}6 : (A \times B \times C \times D \times E \times F) \times B \rightarrow \\
\qquad\qquad\qquad A \times B \times C \times D \times E \times F \\
\hline
\forall\, \mathit{abcdef} : A \times B \times C \times D \times E \times F;\ b : B \bullet \\
\quad \exists\, a : A;\ b_0 : B;\ c : C;\ d : D;\ e : E;\ f : F \mid \mathit{abcdef} = (a, b_0, c, d, e, f) \bullet \\
\qquad \mathit{updateSecondOf}6(\mathit{abcdef}, b) = (a, b, c, d, e, f)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D, E, F] \\
\hline
\mathit{updateThirdOf}6 : (A \times B \times C \times D \times E \times F) \times C \rightarrow \\
\qquad\qquad\qquad A \times B \times C \times D \times E \times F \\
\hline
\forall\, \mathit{abcdef} : A \times B \times C \times D \times E \times F;\ c : C \bullet \\
\quad \exists\, a : A;\ b : B;\ c_0 : C;\ d : D;\ e : E;\ f : F \mid \mathit{abcdef} = (a, b, c_0, d, e, f) \bullet \\
\qquad \mathit{updateThirdOf}6(\mathit{abcdef}, c) = (a, b, c, d, e, f)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D, E, F] \\
\hline
\mathit{updateFourthOf}6 : (A \times B \times C \times D \times E \times F) \times D \rightarrow \\
\qquad\qquad\qquad A \times B \times C \times D \times E \times F \\
\hline
\forall\, \mathit{abcdef} : A \times B \times C \times D \times E \times F;\ d : D \bullet \\
\quad \exists\, a : A;\ b : B;\ c : C;\ d_0 : D;\ e : E;\ f : F \mid \mathit{abcdef} = (a, b, c, d_0, e, f) \bullet \\
\qquad \mathit{updateFourthOf}6(\mathit{abcdef}, d) = (a, b, c, d, e, f)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D, E, F] \\
\hline
\mathit{updateFifthOf}6 : (A \times B \times C \times D \times E \times F) \times E \to \\
\qquad\qquad\qquad A \times B \times C \times D \times E \times F \\
\hline
\forall\, \mathit{abcdef} : A \times B \times C \times D \times E \times F;\ e : E \bullet \\
\quad \exists\, a : A;\ b : B;\ c : C;\ d : D;\ e_0 : E;\ f : F \mid \mathit{abcdef} = (a, b, c, d, e_0, f) \bullet \\
\qquad \mathit{updateFifthOf}6(\mathit{abcdef}, e) = (a, b, c, d, e, f)
\end{array}
$$

$$
\begin{array}{l}
[A, B, C, D, E, F] \\
\hline
\mathit{updateSixthOf}6 : (A \times B \times C \times D \times E \times F) \times F \to \\
\qquad\qquad\qquad A \times B \times C \times D \times E \times F \\
\hline
\forall\, \mathit{abcdef} : A \times B \times C \times D \times E \times F;\ f : F \bullet \\
\quad \exists\, a : A;\ b : B;\ c : C;\ d : D;\ e : E;\ f_0 : F \mid \mathit{abcdef} = (a, b, c, d, e, f_0) \bullet \\
\qquad \mathit{updateSixthOf}6(\mathit{abcdef}, f) = (a, b, c, d, e, f)
\end{array}
$$

We're going to Greenspun up some stuff.

$$\mathit{Ord} ::= \mathit{LT} \mid \mathit{EQ} \mid \mathit{GT}$$

$$
\begin{array}{l}
\mathit{month2Nat} : \mathit{Month} \to \mathbb{N} \\
\hline
\mathit{month2Nat}\ \mathit{January} = 1\ \land \\
\mathit{month2Nat}\ \mathit{February} = 2\ \land \\
\mathit{month2Nat}\ \mathit{March} = 3\ \land \\
\mathit{month2Nat}\ \mathit{April} = 4\ \land \\
\mathit{month2Nat}\ \mathit{May} = 5\ \land \\
\mathit{month2Nat}\ \mathit{June} = 6\ \land \\
\mathit{month2Nat}\ \mathit{July} = 7\ \land \\
\mathit{month2Nat}\ \mathit{August} = 8\ \land \\
\mathit{month2Nat}\ \mathit{September} = 9\ \land \\
\mathit{month2Nat}\ \mathit{October} = 10\ \land \\
\mathit{month2Nat}\ \mathit{November} = 11\ \land \\
\mathit{month2Nat}\ \mathit{December} = 12
\end{array}
$$

$$
\begin{array}{l}
\mathit{InFlag2Nat} : \mathit{InappropriateFlag} \to \mathbb{N} \\
\hline
\mathit{InFlag2Nat}\ \mathit{Zero} = 0\ \land \\
\mathit{InFlag2Nat}\ \mathit{First} = 1\ \land \\
\mathit{InFlag2Nat}\ \mathit{Second} = 2\ \land \\
\mathit{InFlag2Nat}\ \mathit{Third} = 3\ \land \\
\mathit{InFlag2Nat}\ \mathit{Remove} = 4
\end{array}
$$

$$
\begin{array}{|l}
\mathit{Nat2InFlag} : \mathbb{N} \mathrel{⇸} \mathit{InappropriateFlag} \\
\hline
\mathit{Nat2InFlag}\ 0 = \mathit{Zero} \land \\
\mathit{Nat2InFlag}\ 1 = \mathit{First} \land \\
\mathit{Nat2InFlag}\ 2 = \mathit{Second} \land \\
\mathit{Nat2InFlag}\ 3 = \mathit{Third} \land \\
\mathit{Nat2InFlag}\ 4 = \mathit{Remove}
\end{array}
$$

function 42 **leftassoc** $(\_ \otimes \_)$

$$
\begin{array}{|l}
\_ \otimes \_ : \mathbb{P}\,\mathit{Applicant} \times \mathit{Applicant} \to \mathbb{P}\,\mathit{Applicant} \\
\hline
\forall a_1 : \mathit{Applicant};\ as : \mathbb{P}\,\mathit{Applicant} \bullet \\
\quad \exists p_0, p_1 : \mathit{Password}; \\
\quad \mathit{fn}_0, \mathit{fn}_1 : \mathit{FirstName}; \\
\quad \mathit{ln}_0, \mathit{ln}_1 : \mathit{LastName}; \\
\quad \mathit{ad}_0, \mathit{ad}_1 : \mathit{AdminFlag}; \\
\quad e : \mathit{Email}; \\
\quad a_0 : \mathit{Applicant} \mid \\
\qquad a_1 = (e, p_1, \mathit{fn}_1, \mathit{ln}_1, \mathit{ad}_1) \land (a_0 = (e, p_0, \mathit{fn}_0, \mathit{ln}_0, \mathit{ad}_0) \in as) \bullet \\
\qquad\quad as \otimes a_1 = (as \setminus \{a_0\}) \cup \{a_1\}
\end{array}
$$

$$
\begin{array}{|l}
\mathit{updateHostPrefs} : \mathbb{P}\,\mathit{HostPreference} \times \mathit{HostPreference} \to \mathbb{P}\,\mathit{HostPreference} \\
\hline
\forall hp : \mathit{HostPreference};\ hps : \mathbb{P}\,\mathit{HostPreference} \bullet \\
\quad \exists hp_0 : \mathit{HostPreference} \mid \\
\qquad \mathit{firstOf6}\ hp_0 = \mathit{firstOf6}\ hp \bullet \\
\qquad\quad \mathit{updateHostPrefs}(hps, hp) = (hps \setminus \{hp_0\}) \cup \{hp\}
\end{array}
$$

$\mathit{RemoveUser}$

$\Delta \mathit{HomestayDatabase}$
$A? : \mathit{Applicant}$
$E : \mathit{Email}$
$P : \mathit{Password}$
$\mathit{FN} : \mathit{FirstName}$
$\mathit{LN} : \mathit{LastName}$
$\mathit{AF} : \mathit{AdminFlag}$

$(E, P, \mathit{FN}, \mathit{LN}, \mathit{AF}) = A?$
$\mathit{Applicants}' = \mathit{Applicants} \setminus \{A?\}$
$\mathit{Emails}' = \mathit{Emails} \setminus \{E\}$
$\mathit{HostPreferences}' = \mathit{HostPreferences}$
$\mathit{PhoneNumbers}' = \mathit{PhoneNumbers}$
$\mathit{Passwords}' = \mathit{Passwords}$
$\mathit{UserPhoto}' = \mathit{UserPhoto}$
$\mathit{Valid}' = \mathit{Valid}$
$\mathit{Invalid}' = \mathit{Invalid}$
$\mathit{Postings}' = \mathit{Postings}$

2 Login

section *login* **parents** *general*

When we go to create a new account, we need some information from the user. We update all of our sets to reflect the new addition. After they have successfully created an account, they are taken to the main menu.

$$
\begin{array}{|l}
\multicolumn{1}{l}{\_\, \mathit{CreateUserAccount} \,\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_} \\
\Delta \mathit{HomestayDatabase} \\
E? : \mathit{Email} \\
\mathit{FN}? : \mathit{FirstName} \\
\mathit{LN}? : \mathit{LastName} \\
P? : \mathit{Password} \\
\hline
E? \notin \mathit{Emails} \\
\mathit{Emails}' = \mathit{Emails} \cup \{E?\} \\
\mathit{Passwords}' = \mathit{Passwords} \cup \{P?\} \\
\mathit{Applicants}' = \mathit{Applicants} \cup \{(E?, P?, \mathit{FN}?, \mathit{LN}?, \mathit{NotAdmin})\} \\
\mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \\
\mathit{UserPhoto}' = \mathit{UserPhoto} \\
\mathit{Valid}' = \mathit{Valid} \\
\mathit{Invalid}' = \mathit{Invalid} \\
\mathit{Postings}' = \mathit{Postings} \\
\mathit{HostPreferences}' = \mathit{HostPreferences} \\
\hline
\end{array}
$$

$$
\begin{array}{|l}
\multicolumn{1}{l}{\mathit{CreateAdminAccount}} \\
\hline
\Delta \mathit{HomestayDatabase} \\
E? : \mathit{Email} \\
\mathit{FN}? : \mathit{FirstName} \\
\mathit{LN}? : \mathit{LastName} \\
P? : \mathit{Password} \\
\mathit{AD}? : \mathit{AdminFlag} \\
\hline
E? \notin \mathit{Emails} \\
\mathit{AD}? = \mathit{Admin} \\
\mathit{Emails}' = \mathit{Emails} \cup \{E?\} \\
\mathit{Passwords}' = \mathit{Passwords} \cup \{P?\} \\
\mathit{Applicants}' = \mathit{Applicants} \cup \{(E?, P?, \mathit{FN}?, \mathit{LN}?, \mathit{AD}?)\} \\
\mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \\
\mathit{UserPhoto}' = \mathit{UserPhoto} \\
\mathit{Valid}' = \mathit{Valid} \\
\mathit{Invalid}' = \mathit{Invalid} \\
\mathit{Postings}' = \mathit{Posting} \\
\mathit{HostPreferences}' = \mathit{HostPreferences} \\
\hline
\end{array}
$$

A person can create either a user account, or they can create an admin account. The idea is that the admin account isn't something that you can specify, but you must be given a link to sign up for. The link you follow gives the admin flag.

$$\mathit{CreateAccount} == \mathit{CreateUserAccount} \lor \mathit{CreateAdminAccount}$$

To login, a user needs to enter their email and password. At this point they are taken to the main menu.

$$
\begin{array}{|l}
\multicolumn{1}{l}{\mathit{Login}} \\
\hline
\Xi \mathit{HomestayDatabase} \\
E? : \mathit{Email} \\
P? : \mathit{Password} \\
\mathit{FN} : \mathit{FirstName} \\
\mathit{LN} : \mathit{LastName} \\
\mathit{AD} : \mathit{AdminFlag} \\
\mathit{Resp}! : \mathit{Response} \\
\hline
E? \in \mathit{Emails} \\
P? \in \mathit{Passwords} \\
(E?, P?, \mathit{FN}, \mathit{LN}, \mathit{AD}) \in \mathit{Applicants} \\
\mathit{Resp}! = \mathit{LoginSuccessful} \\
\hline
\end{array}
$$

Users can reset their password if they forget it. We take an email address, generate a password token, then dish that off to the email address. The user then finds the email with the reset token/link and proceeds to reset their password.

This is supposed to be one more step of indirection so that the user wont have their password reset at random. Of course, if the email address is already compromised, it doesn't make much difference.

$$
\begin{array}{|l}
\textit{ForgotPassword} \\
\hline
\Delta \textit{HomestayDatabase} \\
E? : \textit{Email} \\
T! : \textit{PToken} \\
\textit{ID} : \mathbb{N} \\
P : \textit{Password} \\
\textit{FN} : \textit{FirstName} \\
\textit{LN} : \textit{LastName} \\
\textit{AD} : \textit{AdminFlag} \\
\hline
E? \in \textit{Emails} \\
(E?, P, \textit{FN}, \textit{LN}, \textit{AD}) \in \textit{Applicants} \\
\textit{ID} = \#\,\textit{Valid} + \#\,\textit{Invalid} + 1 \\
T! = \textit{ID} \mapsto E? \\
\textit{Valid}' = \textit{Valid} \cup \{T!\} \\
\textit{Invalid}' = \textit{Invalid} \\
\textit{Applicants}' = \textit{Applicants} \\
\textit{Emails}' = \textit{Emails} \\
\textit{Passwords}' = \textit{Passwords} \\
\textit{PhoneNumbers}' = \textit{PhoneNumbers} \\
\textit{UserPhoto}' = \textit{UserPhoto} \\
\textit{Postings}' = \textit{Postings} \\
\textit{HostPreferences}' = \textit{HostPreferences} \\
\hline
\end{array}
$$

Once the user has the password token, they can enter their new password. We let them know that the reset was successful.

$\mathit{ResetPassword}$

$\Delta \mathit{HomestayDatabase}$
$P?, P : \mathit{Password}$
$T? : \mathit{PToken}$
$\mathit{Resp}! : \mathit{Response}$
$\mathit{ID} : \mathbb{N}$
$E : \mathit{Email}$
$\mathit{FN} : \mathit{FirstName}$
$\mathit{LN} : \mathit{LastName}$
$\mathit{AD} : \mathit{AdminFlag}$
$\mathit{App}_0, \mathit{App}_1 : \mathit{Applicant}$

$T? \in \mathit{Valid}$
$E = \mathit{second}\ \ T?$
$\mathit{Valid}' = \mathit{Valid} \setminus \{T?\}$
$\mathit{Invalid}' = \mathit{Invalid} \cup \{T?\}$
$\mathit{App}_0 = (E, P, \mathit{FN}, \mathit{LN}, \mathit{AD}) \in \mathit{Applicants}$
$\mathit{App}_1 = (E, P?, \mathit{FN}, \mathit{LN}, \mathit{AD})$
$\mathit{Applicants}' = \mathit{Applicants} \otimes \mathit{App}_1$
$\mathit{Resp}! = \mathit{PasswordResetSuccessful}$
$\mathit{Emails}' = \mathit{Emails}$
$\mathit{Passwords}' = \mathit{Passwords}$
$\mathit{PhoneNumbers}' = \mathit{PhoneNumbers}$
$\mathit{Postings}' = \mathit{Postings}$
$\mathit{HostPreferences}' = \mathit{HostPreferences}$

3 Student

section *student* **parents** *general*

4 Host

section *host* **parents** *general*

The host section allows the user to view and modify preferences specific to hosting a location. There are some general preferences for each host, e.g. smoking, and pets. Then there are preferences for each host location, e.g. price and availability.

4.1 Preferences

$$
\begin{array}{|l}
\multicolumn{1}{l}{\mathit{EditHostSmoking}} \\
\hline
\Delta \mathit{HomestayDatabase} \\
\mathit{H}? : \mathit{HostPreference} \\
\mathit{H} : \mathit{HostPreference} \\
\mathit{S}? : \mathit{Smoking} \\
\hline
\mathit{H}? \in \mathit{HostPreferences} \\
\mathit{H} = \mathit{updateSecondOf}6(\mathit{H}?, \mathit{S}?) \\
\mathit{HostPreferences}' = \mathit{updateHostPrefs}(\mathit{HostPreferences}, \mathit{H}) \\
\mathit{Applicants}' = \mathit{Applicants} \\
\mathit{Emails}' = \mathit{Emails} \\
\mathit{Passwords}' = \mathit{Passwords} \\
\mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \\
\mathit{UserPhoto}' = \mathit{UserPhoto} \\
\mathit{Valid}' = \mathit{Valid} \\
\mathit{Invalid}' = \mathit{Invalid} \\
\mathit{Postings}' = \mathit{Postings} \\
\hline
\end{array}
$$

$\mathit{EditHostPets}$

$\Delta \mathit{HomestayDatabase}$
$H?, H : \mathit{HostPreference}$
$P? : \mathit{Pets}$

$H? \in \mathit{HostPreferences}$
$H = \mathit{updateThirdOf}6(H?, P?)$
$\mathit{HostPreferences}' = \mathit{updateHostPrefs}(\mathit{HostPreferences}, H)$
$\mathit{Applicants}' = \mathit{Applicants}$
$\mathit{Emails}' = \mathit{Emails}$
$\mathit{Passwords}' = \mathit{Passwords}$
$\mathit{PhoneNumbers}' = \mathit{PhoneNumbers}$
$\mathit{UserPhoto}' = \mathit{UserPhoto}$
$\mathit{Valid}' = \mathit{Valid}$
$\mathit{Invalid}' = \mathit{Invalid}$
$\mathit{Postings}' = \mathit{Postings}$

$\mathit{EditHostChildren}$

$\Delta \mathit{HomestayDatabase}$
$H?, H : \mathit{HostPreference}$
$C? : \mathit{Children}$

$H? \in \mathit{HostPreferences}$
$H = \mathit{updateFourthOf}6(H?, C?)$
$\mathit{HostPreferences}' = \mathit{updateHostPrefs}(\mathit{HostPreferences}, H)$
$\mathit{Applicants}' = \mathit{Applicants}$
$\mathit{Emails}' = \mathit{Emails}$
$\mathit{Passwords}' = \mathit{Passwords}$
$\mathit{PhoneNumbers}' = \mathit{PhoneNumbers}$
$\mathit{UserPhoto}' = \mathit{UserPhoto}$
$\mathit{Valid}' = \mathit{Valid}$
$\mathit{Invalid}' = \mathit{Invalid}$
$\mathit{Postings}' = \mathit{Postings}$

$$
\begin{array}{|l}
\multicolumn{1}{l}{\mathit{EditHostDiet}} \\
\hline
\Delta \mathit{HomestayDatabase} \\
H?, H : \mathit{HostPreference} \\
D? : \mathit{Diet} \\
\hline
H? \in \mathit{HostPreferences} \\
H = \mathit{updateFifthOf}6(H?, D?) \\
\mathit{HostPreferences}' = \mathit{updateHostPrefs}(\mathit{HostPreferences}, H) \\
\mathit{Applicants}' = \mathit{Applicants} \\
\mathit{Emails}' = \mathit{Emails} \\
\mathit{Passwords}' = \mathit{Passwords} \\
\mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \\
\mathit{UserPhoto}' = \mathit{UserPhoto} \\
\mathit{Valid}' = \mathit{Valid} \\
\mathit{Invalid}' = \mathit{Invalid} \\
\mathit{Postings}' = \mathit{Postings} \\
\hline
\end{array}
$$

$$
\begin{array}{|l}
\multicolumn{1}{l}{\mathit{EditHostReligion}} \\
\hline
\Delta \mathit{HomestayDatabase} \\
H?, H : \mathit{HostPreference} \\
R? : \mathit{Religion} \\
\hline
H? \in \mathit{HostPreferences} \\
H = \mathit{updateSixthOf}6(H?, R?) \\
\mathit{HostPreferences}' = \mathit{updateHostPrefs}(\mathit{HostPreferences}, H) \\
\mathit{Applicants}' = \mathit{Applicants} \\
\mathit{Emails}' = \mathit{Emails} \\
\mathit{Passwords}' = \mathit{Passwords} \\
\mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \\
\mathit{UserPhoto}' = \mathit{UserPhoto} \\
\mathit{Valid}' = \mathit{Valid} \\
\mathit{Invalid}' = \mathit{Invalid} \\
\mathit{Postings}' = \mathit{Postings} \\
\hline
\end{array}
$$

$$
\begin{array}{l}
\mathit{EditHostPreferences} == \mathit{EditHostSmoking} \lor \\
\quad \mathit{EditHostPets} \lor \\
\quad \mathit{EditHostChildren} \lor \\
\quad \mathit{EditHostDiet} \lor \\
\quad \mathit{EditHostReligion}
\end{array}
$$

4.2 Validation

So long as the start date is before the end date, we consider the Availability to be valid.

$$\begin{array}{|l}
\multicolumn{1}{l}{\textit{ValiDateGood}} \\ \hline
Avail? : Availability \\
AvailResp! : Response \\
Start, End : Date \\
Y_0, Y_1 : Year \\
M_0, M_1 : Month \\
D_0, D_1 : Day \\ \hline
((M_0, D_0, Y_0), (M_1, D_1, Y_1)) = Avail? \\
(Y_0 < Y_1) \lor \\
\quad (Y_0 = Y_1 \land month2Nat\ M_0 < month2Nat\ M_1) \lor \\
\quad (Y_0 = Y_1 \land M_0 = M_1 \land D_0 < D_1) \\
AvailResp! = ValidAvailability \\ \hline
\end{array}$$

If the start date is after the end date, then no bueno.

$$\begin{array}{|l}
\multicolumn{1}{l}{\textit{ValiDateBad}} \\ \hline
Avail? : Availability \\
AvailResp! : Response \\
Start, End : Date \\
Y_0, Y_1 : Year \\
M_0, M_1 : Month \\
D_0, D_1 : Day \\ \hline
((M_0, D_0, Y_0), (M_1, D_1, Y_1)) = Avail? \\
(Y_0 > Y_1) \lor \\
\quad (Y_0 = Y_1 \land month2Nat\ M_0 > month2Nat\ M_1) \lor \\
\quad (Y_0 = Y_1 \land M_0 = M_1 \land D_0 > D_1) \\
AvailResp! = InvalidAvailability \\ \hline
\end{array}$$

$$ValiDate == ValiDateGood \lor ValiDateBad$$

4.3 Postings

To create a new posting, the host enters all the relevant information for hosting

$NewPosting$

$$\begin{array}{l}
\Delta HomestayDatabase \\
ValiDate \\
H? : HostPreference \\
Addr? : Address \\
E? : ExactOrArea \\
WIDTH? : \mathbb{N} \\
LENGTH? : \mathbb{N} \\
PR? : Price \\
Avail? : Availability \\
PH? : Photo \\
P : Posting \\
AI : AddressInfo \\
S : SizeOfRoom \\
AvailResp! : Response
\end{array}$$

$$\begin{array}{l}
AvailResp! = ValidAvailability \\
AI = Addr? \mapsto E? \\
S = WIDTH? \mapsto LENGTH? \\
P = (H?, AI, S, PR?, Avail?, PH?) \\
Postings' = Postings \cup \{P\} \\
Applicants' = Applicants \\
Emails' = Emails \\
Passwords' = Passwords \\
PhoneNumbers' = PhoneNumbers \\
UserPhoto' = UserPhoto \\
Valid' = Valid \\
Invalid' = Invalid \\
HostPreferences' = HostPreferences
\end{array}$$

Deleting a post removes the post from Postings

$$
\begin{array}{|l}
\multicolumn{1}{l}{\textit{DeletePosting}} \\
\hline
\Delta \mathit{HomestayDatabase} \\
P? : \mathit{Posting} \\
\hline
P? \in \mathit{Postings} \\
\mathit{Postings}' = \mathit{Postings} \setminus \{P?\} \\
\mathit{Applicants}' = \mathit{Applicants} \\
\mathit{Emails}' = \mathit{Emails} \\
\mathit{Passwords}' = \mathit{Passwords} \\
\mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \\
\mathit{UserPhoto}' = \mathit{UserPhoto} \\
\mathit{Valid}' = \mathit{Valid} \\
\mathit{Invalid}' = \mathit{Invalid} \\
\mathit{HostPreferences}' = \mathit{HostPreferences} \\
\hline
\end{array}
$$

5 Administrator

section *admin* **parents** *general*

Administrator login only needs to act like an inbox with ”flag as inappropriate” or ”accept/reject match” messages from students. The administrator will have a link to the group chat that is flagged and can add messages or end the chat if it is inappropriate. If the student and host decide to chat, once they send in their accept/reject match to the admin, the admin will have final approval to accept or reject the match.

$$
\begin{array}{l}
\mathit{MatchFlag} ::= \\
\quad \mathit{Accept} \\
\quad \mid \mathit{Reject}
\end{array}
$$

$$
\begin{array}{l}
\mathit{Message} ::= \\
\quad \mathit{OneAccept} \\
\quad \mid \mathit{TwoAccepts} \\
\quad \mid \mathit{NoAccepts} \\
\quad \mid \mathit{Inappropriate}
\end{array}
$$

$$
\begin{array}{|l}
\mathit{AdminMatch} : \mathit{Message} \nrightarrow \mathit{MatchFlag} \\
\hline
\mathit{AdminMatch}\ \mathit{OneAccept} = \mathit{Reject} \land \\
\mathit{AdminMatch}\ \mathit{NoAccepts} = \mathit{Reject} \land \\
\mathit{AdminMatch}\ \mathit{TwoAccepts} = \mathit{Accept}
\end{array}
$$

$$\begin{array}{l}
\mathit{AdminInappropriate} \\
\hline
\Delta \mathit{HomestayDatabase} \\
M? : \mathit{Message} \\
E? : \mathit{Email} \\
I : \mathit{IFlag} \\
\mathit{In}, \mathit{In2} : \mathit{InappropriateFlag} \\
\hline
M? = \mathit{Inappropriate} \\
\{E? \mapsto \mathit{In}\} = \{E?\} \lhd \mathit{IFlags} \\
\mathit{In} \neq \mathit{Remove} \\
\mathit{In2} = \mathit{Nat2InFlag}((\mathit{InFlag2Nat}\ \mathit{In}) + 1) \\
\mathit{IFlags}' = \mathit{IFlags} \oplus \{E? \mapsto \mathit{In2}\} \\
\mathit{In2} \neq \mathit{Remove} \\
\mathit{Applicants}' = \mathit{Applicants} \\
\mathit{Emails}' = \mathit{Emails} \\
\mathit{Passwords}' = \mathit{Passwords} \\
\mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \\
\mathit{UserPhoto}' = \mathit{UserPhoto} \\
\mathit{Valid}' = \mathit{Valid} \\
\mathit{Invalid}' = \mathit{Invalid}
\end{array}$$

$$\begin{array}{l}
\mathit{InappropriateUser} \\
\hline
\Delta \mathit{HomestayDatabase} \\
M? : \mathit{Message} \\
E? : \mathit{Email} \\
I : \mathit{IFlag} \\
\mathit{In}, \mathit{In2} : \mathit{InappropriateFlag} \\
\mathit{RemoveUser} \\
\hline
M? = \mathit{Inappropriate} \\
\{E? \mapsto \mathit{In}\} = \{E?\} \lhd \mathit{IFlags} \\
\mathit{In} \neq \mathit{Remove} \\
\mathit{In2} = \mathit{Nat2InFlag}((\mathit{InFlag2Nat}\ \mathit{In}) + 1) \\
\mathit{IFlags}' = \mathit{IFlags} \oplus \{E? \mapsto \mathit{In2}\} \\
\mathit{In2} = \mathit{Remove}
\end{array}$$

6 Group Chat

section $group_chat$ **parents** $general$

7 Profile

section *profile* **parents** *general*

The profile section has functionality to update/change basic user information: First Name, Last Name, User Photo, Email, and Phone Number.

EditFirstName

$$
\begin{array}{|l}
\Delta \mathit{HomestayDatabase} \\
\mathit{FN}?, \mathit{FN} : \mathit{FirstName} \\
\mathit{App}_0?, \mathit{App}_1 : \mathit{Applicant} \\
\mathit{LN} : \mathit{LastName} \\
E : \mathit{Email} \\
P : \mathit{Password} \\
\mathit{AD} : \mathit{AdminFlag} \\
\hline
E \in \mathit{Emails} \\
P \in \mathit{Passwords} \\
\mathit{App}_0? \in \mathit{Applicants} \\
\mathit{App}_1 = \mathit{updateThirdOf}5(\mathit{App}_0?, \mathit{FN}?) \\
\mathit{Applicants}' = \mathit{Applicants} \otimes \mathit{App}_1 \\
\mathit{Emails}' = \mathit{Emails} \\
\mathit{Passwords}' = \mathit{Passwords} \\
\mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \\
\mathit{Valid}' = \mathit{Valid} \\
\mathit{Invalid}' = \mathit{Invalid} \\
\mathit{HostPreferences}' = \mathit{HostPreferences} \\
\mathit{Postings}' = \mathit{Postings}
\end{array}
$$

$\mathit{EditLastName}$

$$
\begin{array}{|l}
\Delta \mathit{HomestayDatabase} \\
\mathit{FN} : \mathit{FirstName} \\
\mathit{LN}, \mathit{LN}? : \mathit{LastName} \\
E : \mathit{Email} \\
P : \mathit{Password} \\
\mathit{AD} : \mathit{AdminFlag} \\
\mathit{App}_0, \mathit{App}_1 : \mathit{Applicant} \\
\hline
E \in \mathit{Emails} \\
P \in \mathit{Passwords} \\
\mathit{App}_0 = (E, P, \mathit{FN}, \mathit{LN}, \mathit{AD}) \in \mathit{Applicants} \\
\mathit{App}_1 = (E, P, \mathit{FN}, \mathit{LN}?, \mathit{AD}) \\
\mathit{Applicants}' = \mathit{Applicants} \otimes \mathit{App}_1 \\
\mathit{Emails}' = \mathit{Emails} \\
\mathit{Passwords}' = \mathit{Passwords} \\
\mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \\
\mathit{UserPhoto}' = \mathit{UserPhoto} \\
\mathit{Valid}' = \mathit{Valid} \\
\mathit{Invalid}' = \mathit{Invalid} \\
\mathit{HostPreferences}' = \mathit{HostPreferences} \\
\mathit{Postings}' = \mathit{Postings}
\end{array}
$$

$\mathit{EditPhoto}$

$$
\begin{array}{|l}
\Delta \mathit{HomestayDatabase} \\
\mathit{FN} : \mathit{FirstName} \\
\mathit{LN} : \mathit{LastName} \\
P : \mathit{Password} \\
\mathit{UP}?, \mathit{UP} : \mathit{UserPhoto} \\
E : \mathit{Email} \\
\mathit{AD} : \mathit{AdminFlag} \\
\hline
\exists\, \mathit{App} : \mathit{Applicant} \mid \mathit{App} = (E, P, \mathit{FN}, \mathit{LN}, \mathit{AD}) \in \mathit{Applicants} \land \mathit{UP} \in \mathit{UserPhoto} \bullet \\
\mathit{UserPhoto}' = \mathit{UserPhoto} \setminus \{\mathit{UP}\} \land \mathit{UserPhoto}' = \mathit{UserPhoto} \cup \{\mathit{UP}?\} \\
\mathit{Applicants}' = \mathit{Applicants} \\
\mathit{Emails}' = \mathit{Emails} \\
\mathit{Passwords}' = \mathit{Passwords} \\
\mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \\
\mathit{Valid}' = \mathit{Valid} \\
\mathit{Invalid}' = \mathit{Invalid} \\
\mathit{HostPreferences}' = \mathit{HostPreferences} \\
\mathit{Postings}' = \mathit{Postings}
\end{array}
$$

$\mathit{EditEmail}$

$\Delta \mathit{HomestayDatabase}$
$\mathit{FN} : \mathit{FirstName}$
$\mathit{LN} : \mathit{LastName}$
$P : \mathit{Password}$
$E?, E : \mathit{Email}$
$\mathit{AD} : \mathit{AdminFlag}$

$\exists\, \mathit{App}_0, \mathit{App}_1 : \mathit{Applicant} \mid \mathit{App}_0 = (E, P, \mathit{FN}, \mathit{LN}, \mathit{AD}) \in \mathit{Applicants} \wedge$
$\mathit{App}_1 = (E?, P, \mathit{FN}, \mathit{LN}, \mathit{AD}) \notin \mathit{Applicants} \bullet$
$\mathit{Applicants}' = \mathit{Applicants} \setminus \{\mathit{App}_0\} \wedge \mathit{Applicants}' = \mathit{Applicants} \cup \{\mathit{App}_1\}$
$\mathit{Emails}' = \mathit{Emails}$
$\mathit{Passwords}' = \mathit{Passwords}$
$\mathit{PhoneNumbers}' = \mathit{PhoneNumbers}$
$\mathit{UserPhoto}' = \mathit{UserPhoto}$
$\mathit{Valid}' = \mathit{Valid}$
$\mathit{Invalid}' = \mathit{Invalid}$
$\mathit{HostPreferences}' = \mathit{HostPreferences}$
$\mathit{Postings}' = \mathit{Postings}$

$\mathit{EditPhone}$

$\Delta \mathit{HomestayDatabase}$
$\mathit{FN} : \mathit{FirstName}$
$\mathit{LN} : \mathit{LastName}$
$E : \mathit{Email}$
$P : \mathit{Password}$
$\mathit{AD} : \mathit{AdminFlag}$
$\mathit{PNUM}?, \mathit{PNUM} : \mathit{Phone}$

$\exists\, \mathit{App} : \mathit{Applicant} \mid \mathit{App} = (E, P, \mathit{FN}, \mathit{LN}, \mathit{AD}) \in \mathit{Applicants} \wedge \mathit{PNUM} \in \mathit{PhoneNumbers} \bullet$
$\mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \setminus \{\mathit{PNUM}\} \wedge \mathit{PhoneNumbers}' = \mathit{PhoneNumbers} \cup \{\mathit{PNUM}?\}$
$\mathit{Applicants}' = \mathit{Applicants}$
$\mathit{Emails}' = \mathit{Emails}$
$\mathit{Passwords}' = \mathit{Passwords}$
$\mathit{UserPhoto}' = \mathit{UserPhoto}$
$\mathit{Valid}' = \mathit{Valid}$
$\mathit{Invalid}' = \mathit{Invalid}$
$\mathit{HostPreferences}' = \mathit{HostPreferences}$
$\mathit{Postings}' = \mathit{Postings}$