

# Homestay App Specification

## Types

Base classes from requirement's class diagram:

$[Admin, Profile, File, DateRange, Allergie]$   
 $Response ::= success \mid error$   
 $ProfileType ::= student \mid host$   
 $FamilyStructure ::= couple \mid couplew/kids \mid singleparent \mid gaycouple \mid gaycouplew/kids \mid other$   
 $Pet ::= cat \mid dog \mid chicken \mid birds \mid snakes \mid fish \mid other$

## The System

The system must hold a schema by itself with the purpose of declaring its elements as well as its invariants:

*HOMESTAYAPP*

$\exists STUDENTS$

$\exists HOSTS$

$managers : \mathbb{P} Admin$

$requests : Profile \rightarrow Profile$

$authorized\_requests : Profile \rightarrow Profile$

$students \subseteq Profile$

$hosts \subseteq Profile$

$students \cap Hosts = \emptyset$

$\text{dom } requests \subset students$

$\text{ran } requests \subset hosts$

$requests \cap authorized\_requests = \emptyset$

$\forall s : students \mid s \in \text{dom } Requests \bullet \#(s \triangleleft Requests) \leq 3$

## Actors

*STUDENTS*

$students : \mathbb{P}(Profile \rightarrow ((\mathbb{N} \rightarrow FamilyStructure) \times \mathbb{P}(\mathbb{N} \rightarrow Pet) \times \mathbb{P} Allergie))$

*HOSTS*

$hosts : \mathbb{P}(Profile \rightarrow (\mathbb{P} Pet \times \mathbb{P} FamilyStructure \times \mathbb{P} Allergie))$

## Manual Selection

*MANUAL*  
 $\Xi$ *HOMESTAYAPP*

## Selection Wizard

*WIZARD*  
 $\Xi$ *HOMESTAYAPP*

## General Operations

### Operations for both Student and Host

*CreateNewProfile*  
 $\Delta$ *STUDENTS*  
 $\Delta$ *HOSTS*  
 $type? : ProfileType$   
 $p? : Profile$   
 $family\_str\_if\_host? : FamilyStructure$   
 $pets\_if\_host? : \mathbb{P} Pet$   
 $family\_str\_pref\_if\_std? : \mathbb{P}(\mathbb{N} \rightarrow FamilyStructure)$   
 $pet\_pref\_if\_std? : \mathbb{P} Pet$   
 $allr? : \mathbb{P} Allergier! : Response$   
 $(type? = student \wedge$   
     $students' = students \cup \{p? \mapsto (family\_str\_pref\_if\_std?, pet\_pref\_if\_std?, allr?)\} \wedge$   
     $hosts' = hosts \wedge r! = success) \vee$   
 $(type? = host \wedge$   
     $hosts' = hosts \cup \{p? \mapsto (pets\_if\_host?, family\_str\_if\_host?, allr?)\} \wedge$   
     $students' = students \wedge r! = success)$

When creating a profile the system doesn't yet know if the request is for a student profile or a host profile, that's why we need *type?*.

### *UpdateProfile*

$\Delta STUDENTS$

$\Delta HOSTS$

$p? : Profile$

$family\_str\_if\_host? : FamilyStructure$

$pets\_if\_host? : \mathbb{P} Pet$

$family\_str\_pref\_if\_std? : \mathbb{P}(\mathbb{N} \rightarrow FamilyStructure)$

$pet\_pref\_if\_std? : \mathbb{P} Pet$

$allr? : \mathbb{P} Allergier! : Response$

$(p? \in \text{dom } students \wedge$

$students' = students \cup \{p? \mapsto (family\_str\_pref\_if\_std?, pet\_pref\_if\_std?, allr?)\} \wedge$

$hosts' = hosts \wedge r! = success) \vee$

$(p? \in \text{dom } students \wedge$

$hosts' = hosts \cup \{p? \mapsto (pets\_if\_host?, family\_str\_if\_host?, allr?)\} \wedge$

$students' = students \wedge r! = success) \vee$

$(p? \notin \text{dom } students \wedge p? \notin \text{dom } hosts \wedge$

$students' = students$

$hosts' = hosts$

$r! = error)$

When updating, we can tell from  $p?$  if its about students or hosts. Either way, this operation overwrites the data previously inserted for any information about students and/or hosts.

### *DeleteProfile*

$\Delta STUDENTS$

$\Delta HOSTS$

### *ViewRequests*

$\Xi HOMESTAYAPP$

## Operations related to Student

### *ManualRequest*

$\Delta HOMESTAYAPP$

$\Xi MANUAL$

### *WizardRequest*

$\Delta HOMESTAYAPP$

$\Xi WIZARD$

*SetPetPreference*  
 $\Delta STUDENTS$

*SetFamilyStructurePreference*  
 $\Delta STUDENTS$

*SetSmokingPreference*  
 $\Delta STUDENTS$

*SetAllergies*  
 $\Delta STUDENTS$

### Operations related to Host

*AcceptRequest*  
 $\Delta HOMESTAYAPP$

*SetPets*  
 $\Delta HOSTS$

*SetFamilyStructure*  
 $\Delta HOSTS$

### Operations related to Admin

*ViewPendingRequests*  
 $\Xi HOMESTAYAPP$

*AuthorizeRequest*  
 $\Delta HOMESTAYAPP$

*ArrangeMeeting*  
 $\Delta HOMESTAYAPP$