

0. システムを起動する

1. リストを見る

2. 学生を見る

3. 単位を修正する

4. 学生を追加する

5. 学生を削除する

6. システムを終了する

#### 【ユースケース：1. リストを見る】

- ・概要：ユーザがシステムに登録されている学生の一覧を見る。
- ・事前条件：システムが起動済みでメニューが表示されていること。
- ・トリガー：ユーザがメニューから対応する数字を入力する。
- ・基本フロー：
  1. ユーザがメニューから「リストを見る」に対応する数字を入力する。
  2. システムは登録されている全ての学生の情報をリストにして、画面に表示する。  
ここで学生の情報とは、[学籍番号，名前，単位数，学生の種別] を1行とする情報である。  
また，システムで扱う学生は，現状以下の4種類が存在する：
    - 正規学生：学籍番号，名前，単位数の情報を保持する。
    - 留学生：正規学生の情報に加えて，出身国，国費・私費の別を保持する。
    - 社会人学生：正規学生の情報に加えて，勤務先の情報を保持する。
    - ロボット学生：正規学生の情報に加えて，ベンダ，アルゴリズムの情報を管理する。
  3. ユーザは画面上の学生のリストを確認する。
- ・代替フロー：特に無し
- ・事後条件：画面に学生のリストが表示されていること。

#### 【ユースケース：3. 単位を修正する】

- ・概要：ユーザが学籍番号と単位数を指定して，その学生の単位数を修正する。
- ・事前条件：システムが起動済みでメニューが表示されていること。
- ・トリガー：ユーザがメニューから対応する数字を入力する。
- ・基本フロー：
  1. ユーザがメニューから「単位を修正する」に対応する数字を入力する。
  2. システムは，どの学生の単位を修正するか，学籍番号を入力するようユーザに尋ねる。
  3. ユーザは単位を修正したい学生の学籍番号を入力する。
  4. システムは該当する学生の学籍番号，名前，現在の単位数を表示する。
  5. システムは該当する学生の単位を何単位に修正するかユーザに尋ねる。
  6. ユーザは修正する単位数を入力する。
  7. システムは該当する学生の単位数を，入力された単位数に更新する。
  8. システムは該当する学生の学籍番号，名前，修正後の単位数を表示する。
  9. システムは更新された学生リストの内容をデータファイルにセーブする。
- ・代替フロー：
  - 3a. 3でユーザが入力した学籍番号が存在しなければ，システムは存在しない旨を表示し，メニューに戻る。
  - 6a. 6でユーザが入力した単位数が負の値の場合，システムはエラーを表示し，メニューに戻る。
- ・事後条件：該当学生の単位が修正されていること。

#### 【ユースケース：4. 学生を追加する】

- ・概要：ユーザが新たに学生の情報を追加登録する。
- ・事前条件：システムが起動済みでメニューが表示されていること。
- ・トリガー：ユーザがメニューから対応する数字を入力する。
- ・基本フロー：
  1. ユーザがメニューから「学生を追加する」に対応する数字を入力する。
  2. システムは，登録する学生の学籍番号をユーザに尋ねる。
  3. ユーザが学籍番号を入力する。
  4. システムは，登録する学生の名前をユーザに尋ねる。
  5. ユーザが名前を入力する。
  6. システムは，登録する学生の単位数をユーザに尋ねる。
  7. ユーザが学生種別を入力する。種別は(0: 通常学生，1: 留学生，2: 社会人学生，3: ロボット学生)のように指定する。
  8. ユーザは学生種別を入力する。
  9. システムは入力された学生種別に応じて，追加の情報の入力を促す。
    - 9.1 留学生の場合，国籍と国費・私費の別
    - 9.2 社会人の場合，勤務先
    - 9.3 ロボットの場合，ベンダとアルゴリズム。
  10. システムは入力された情報に基づき学生情報を作成し，学生リストに追加する。
  11. システムは更新された学生リストの内容をデータファイルにセーブする。
- ・代替フロー：
  - 3a. 3でユーザが入力した学籍番号が既に存在していれば，システムは既存の学生が存在しない旨を表示し，追加をキャンセルしてメニューに戻る。
  - 6a. 6でユーザが入力した単位数が負の値の場合，システムはエラーを表示し，メニューに戻る。
  - 7a. 7でユーザが入力した種別が存在しない場合，システムはエラーを表示し，メニューに戻る。
- ・事後条件：該当学生がシステムの学生リストに登録されていること。

#### 【ユースケース：0. システムを起動する】

- ・概要：指定されたファイルから学生データをロードし、メニューを表示する
- ・事前条件：特になし
- ・トリガー：プログラムが実行される
- ・基本フロー：
  1. 指定されたファイルから学生データを読み込み、リストに格納する。
  2. ユーザの要求を知るため、メニューを表示する。
- ・代替フロー：指定されたファイルが存在しない場合、ユーザーに警告しメニューに戻る。
- ・事後条件：メニューが表示されていること、学生データがロードされていること。

#### 【ユースケース：2. 学生を見る】

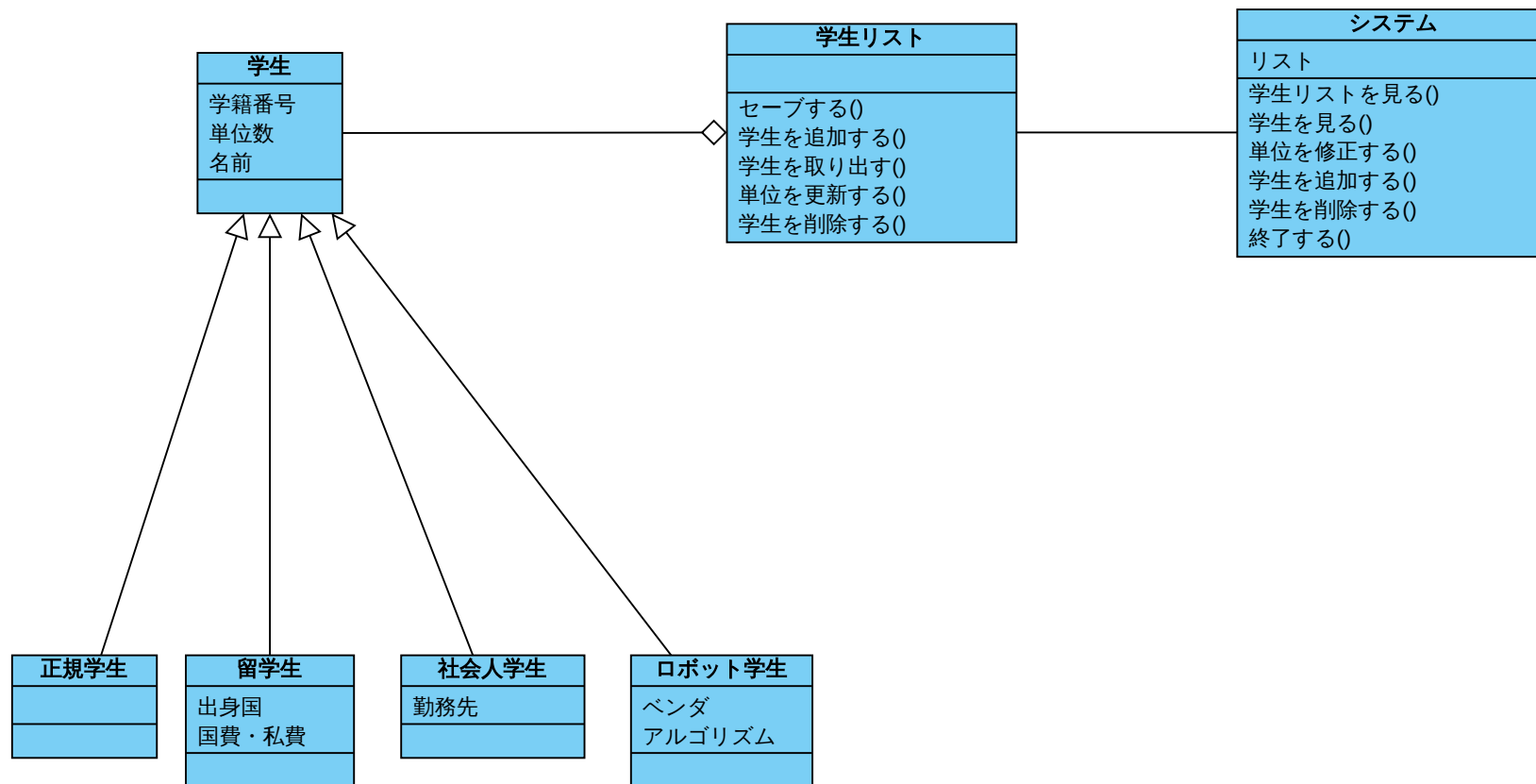
- ・概要：ユーザが学籍番号を指定して、その学生の詳細情報を確認する。
- ・事前条件：システムが起動済みで、メニューが表示されていること。
- ・トリガー：ユーザがメニューから対応する数字を入力する。
- ・基本フロー：
  1. ユーザがメニューから「学生を見る」に対応する数字を入力する。
  2. システムは、どの学生の情報を参照するか、学籍番号を入力するようユーザに尋ねる。
  3. ユーザは参照したい学生の学籍番号を入力する。
  4. システムは該当する学生の学籍番号・名前・現在の単位数を表示する。  
学生が留学生の場合は出身国と国費・私費の別、  
社会人学生の場合は勤務先  
ロボット学生の場合はベンダ・アルゴリズムも表示する。
  5. ユーザは画面上の学生情報を確認する。
- ・代替フロー：特になし
- ・事後条件：画面に学生の詳細情報が表示されていること

#### 【ユースケース：5. 学生を削除する】

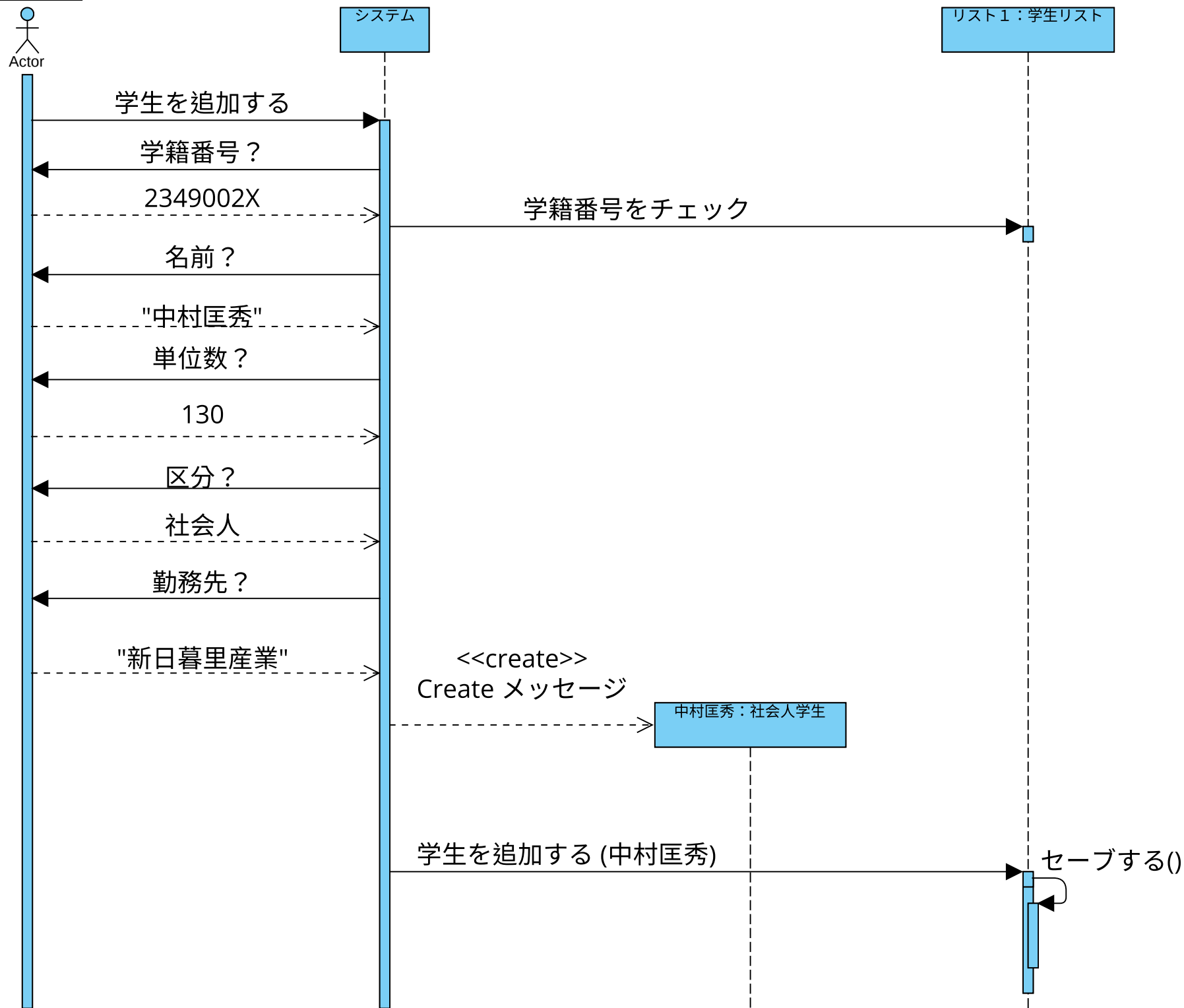
- ・概要：ユーザが学籍番号を指定して、その学生の情報を削除する。
- ・事前条件：システムが起動済みでメニューが表示されていること。
- ・トリガー：ユーザがメニューから対応する数字を入力する。
- ・基本フロー：
  1. ユーザがメニューから「学生を削除する」に対応する数字を入力する
  2. システムは削除する学生の学籍番号をユーザに尋ねる。
  3. ユーザが学籍番号を入力する。
  4. システムは入力された学籍番号に対応する学生をリストから削除する。
  5. システムは更新された学生リストの内容をデータファイルにセーブする。
- ・代替フロー：
  - 4a. 4で指定された学生がリストにいない場合、ユーザに通知しメニューに戻る。
- ・事後条件：該当学生がシステムの学生リストから削除されていること。

#### 【ユースケース：6. システムを終了する】

- ・概要：プログラムを終了する。
- ・事前条件：システムが起動済みでメニューが表示されていること。
- ・トリガー：ユーザがメニューから対応する数字を入力する。
- ・基本フロー：
  1. ユーザがメニューから「システムを終了する」に対応する数字を入力する。
  2. システムは終了する旨を通知し、プログラムを終了する。
- ・代替フロー：特になし
- ・事後条件：プログラムとともにシステムが終了していること。



## sd 4. 学生を追加する



sd 3. 単位を修正する

