K3Requirements

```
Model K3Requirements extends K3DataModel begin
  // Top level items in the system
  refined system:
end K3Requirements;
```

```
system begin
  // k3data Requirements
  partial Model D1;
  partial Model D2;
  partial Model D3;
  partial Model D4;
  partial Model D5;
  partial Model D6;
  partial Model D7;
  partial Model D8;
  partial Model D9;

  // k3ucs Requirements
  partial Model R1;
  partial Model R2;
  partial Model R3;
  partial Model R4;
  partial Model R5;
  partial Model R6;
  partial Model R7;
  partial Model R8;
  partial Model R9;
  partial Model R10;
  partial Model R11;
  partial Model R12;
  partial Model R13;
  partial Model R14;
  partial Model R15;
  partial Model R16;
  partial Model R17;
  partial Model R18;
  partial Model R19;
  partial Model R20;
  partial Model R21;
  partial Model R22;
  partial Model R23;
  partial Model R24;
  partial Model R25;
  partial Model R26;
end system;
```

```
system.D1 begin
  Requirement d1
    "A class shall be either a non-clinical class or a
     clinical class"
    is ensure °Class = ClinicalClass union NonClinicalClass;
end system.D1;
```

```
system.D2 begin
  Requirement ~d2
    "A non-clinical class shall specify the course name, lecture room requirements
     and instructor needs"
  : concretisation is dd2;

  Requirement ~dd2  // Implemented in class definitions
    "A non-clinical class shall specify the class name, lecture room needs
     and instructor needs";
end system.D2;
```

```
system.D3 begin
  Requirement ~d3
    "A clinical class shall specify the course name, lecture room requirements,
     clinical site needs,  lecture instructor needs  and clinical lab instructor
     needs"
  : concretisation is dd3;

  Requirement ~dd3  // Implemented in class definitions
    "A clinical class shall specify the class name, lecture room needs, clinical
     site needs,  lecture instructor needs  and clinical lab instructor needs";
end system.D3;
```

```
system.D4 begin
  Requirement ~d4
    "A Program of Study shall consist of a program name and listing of required
     classes (both clinical and non-clinical) that must be completed"
  : concretisation is dd4;

  Requirement ~dd4   // Implemented in class definitions
    "A Program of Study definition shall include a name and listing of
     required classes (both clinical and non-clinical) that must be
     completed by students enrolled in that Program of Study";
end system.D4;
```

```
system.D5 begin
  Requirement ~d5
    "The report of needed classes shall include (but not be limited
     to) classes to be offered, number of sections needed, number of
     labs needed, and room types needed"
  : concretisation is dd5;

  Requirement dd5
    "The report of needed classes shall include (but not be limited
     to) classes to be offered by the Nursing Department, with the
     corresponding number of sections needed, number of labs needed,
     and room types needed";

  EReport     eReportOfNeededClasses               is deferred;

  Information infoOnClassesToBeOfferedByNursingDep is deferred;
  Information infoOnNbSectionsNeededByNursingDep   is deferred;
  Information infoOnNbLabsNeededByNursingDep       is deferred;
  Information infoOnRoomTypesNeededByNursingDep    is deferred;

  dd5 is when eReportOfNeededClasses ensure
  { infoOnClassesToBeOfferedByNursingDep,
    infoOnNbSectionsNeededByNursingDep,
    infoOnNbLabsNeededByNursingDep,
    infoOnRoomTypesNeededByNursingDep
  } in eReportOfNeededClasses.contents;
end system.D5;
```

```
system.D6 begin
  Requirement ~d6
    "Classes for a given cohort shall not conflict with regards to
     the time and day that they are offered"
  : concretisation is dd6;

  Requirement dd6
    "The sequence of classes of any student of any cohort shall not
     conflict with regards to the time and day that they are offered"
    is for all cohort            of system.cohorts
       for all student           of cohort
       for all section1,section2 of student.sequenceOfClasses.sections
         such that section1.identity <> section2.identity
       ensure not overlap (section1.beginning, section1.°duration,
                           section2.beginning, section2.°duration);
end system.D6;
```

```
system.D7 begin
  Requirement ~d7
    "A clinical lab section shall include the clinical site name,
     the class  instructor, day and time of the lab"
  : concretisation is dd7;

  Requirement ~dd7   // Implemented in class definitions
    "A clinical lab section shall include the clinical site name,
     the class instructor, day and time of the section";
end system.D7;
```

```
system.D8 begin
  Requirement ~d8    // Implemented in class definitions
    "Clinical site information shall include (but not be limited to) the name
     of the site, contact person and contact information";
end system.D8;
```

```
system.D9 begin
   Requirement ~d9   // Implemented in class definitions
     "The system shall contain contact information (e-mail and phone number)
      for all people relevant to the system  including (but not limited to)
      staff members,  students,  lecture instructors,  clinical lab
      instructors and clinical site administrators";
end system.D9;
```

```
system.R1 begin
  Requirement ~r1
    "Program Administrators and Nursing Staff Members shall be able
     to add clinical classes or sections to a sequence of classes"
  : concretisation is {rr1a, rr1b, rr1c, rr1d};

  Requirement rr1a
    "Each Program Administrator, each Nursing Staff Member shall
     be able to add new sections to the system";
  Requirement rr1b
    "Each Program Administrator, each Nursing Staff Member shall
     be able to add new sequences of classes to the system";
  Requirement rr1c
    "Each Program Administrator, each Nursing Staff Member shall
     be able to add any clinical classes in the system to any
     sequence of classes in the system"
  Requirement rr1d
    "Each Program Administrator, each Nursing Staff Member shall
     be able to add any sections in the system to any sequence of
     classes in the system";

  // rr1a
  Event eAddSections begin
    occurrence is deferred;
    Section {} sections;
  end eAddSections;

  rr1a is
    for all x of PA union NSM
    after eAddSections
    assert x can achieve
      bop.sections in system.sections;

  // rr1b
  Event eAddSequencesOfClasses begin
    occurrence is deferred;
    SequenceOfClasses {} sequencesOfClasses;
  end eAddSequencesOfClasses;

  rr1b is
    for all x of PA union NSM
    after eAddSequencesOfClasses
    assert x can achieve
      bop.sequencesOfClasses in system.sequencesOfClasses;
  // rr1c
  Event eAddClinicalClassesToSequence begin
    occurrence is deferred;
    ClinicalClass {}  clinicalClasses : Guarantee g1 is
      when occurrence ensure me in system.clinicalClasses;
    SequenceOfClasses sequenceOfClasses : Guarantee g2 is
      when occurrence ensure me in system.sequencesOfClasses;
  end eAddClinicalClassesToSequence;

  rr1c is
    for all x of PA union NSM
    after eAddClinicalClassesToSequence
    assert x can achieve
      bop.clinicalClasses in bop.sequenceOfClasses.classes;

  // rr1d
  Event eAddSectionsToSequence begin
    occurrence is deferred;
    Section {} sections : Guarantee g1 is
      when occurrence ensure me in system.sections;
    SequenceOfClasses sequenceOfClasses : Guarantee g2 is
      when occurrence ensure me in system.sequencesOfClasses;
  end eAddSectionsToSequence;

  rr1d is
    for all x of PA union NSM
    after eAddSectionsToSequence
    assert x can achieve
      bop.sections in bop.sequenceOfClasses.addedSections;
end system.R1;
```

```
system.R2 begin
  Requirement ~r2
    "Program Administrators and Nursing Staff Members shall be able to add
     a new cohort to the system identified by start month and year"
  : concretisation is {rr2a, rr2b};
  // ..................................................................
  Requirement rr2a
    "Each Program Administrator or Nursing Staff Member shall be able to add
     new cohorts to the system";

  Event eAddCohorts begin
    occurrence is deferred;
    Cohort {} cohorts;
  end eAddCohorts;

  rr2a is
    for all x of PA union NSM
    after eAddCohorts within shortDelay
    assert x can achieve bop.cohorts in system.cohorts;
  // ..................................................................
  Requirement rr2b
    "Cohorts shall be identifiable by start year, start month and program of study"
    is for all c1, c2 of system.cohorts such that c1 <> c2
       ensure c1.startMonth     <> c2.startMonth
           or c1.programOfStudy <> c2.programOfStudy;
end system.R2;
```

```
system.R3 begin
  Requirement ~r3
    "Program Administrators and Nursing Staff Members shall be able to add
     new classes for the nursing department into the system"
  : concretisation is {rr3a, rr3b};
  // .............................................................................
  Requirement rr3a
    "Each Program Administrator or Nursing Staff Member shall be able to add
     new classes into the system";

  Event eAddClasses begin
    occurrence is deferred;
    °Class {} classes;
  end eAddClasses;

  rr3a is
    for all x of PA union NSM
    after eAddClasses within shortDelay
    assert x can achieve bop.classes in system.classes;
  // .............................................................................
  Requirement rr3b
    "Each Program Administrator or Nursing Staff Member shall be able to add
     classes in the system to the nursing department";

  Event eAddClassesToNursingDepartment begin
    occurrence is deferred;
    °Class {} classes
    : Guarantee g1 is when occurrence ensure me in system.classes;
  end eAddClassesToNursingDepartment;

  rr3b is
    for all x of PA union NSM
    after eAddClassesToNursingDepartment within shortDelay
    assert x can achieve bop.classes in system.nursingDepartment.classes;
end system.R3;
```

```
system.R4 begin
  Requirement ~r4
    "Program Administrators/Nursing Staff Members shall be able to create
     a new Program of Study"
  : concretisation is rr4;

  Requirement rr4
    "Each Program Administrator or Nursing Staff Member shall be able to
     create new Programs of Study";

  Event eNewProgramsOfStudy begin
    occurrence is deferred;
    ProgramOfStudy {} programs;
  end eNewProgramOfStudy;

  rr4 is
    for all x of PA union NSM
    after eNewProgramsOfStudy within shortDelay
    assert x can achieve bop.programs in system.programs;
end system.R4;
```

```
system.R5 begin
  Requirement ~r5
    "Program Administrators and Nursing Staff Members shall have the ability
     to specify which classes are required for a Program of Study"
  : concretisation is rr5;

  Requirement rr5 is
    "Each Program Administrator or Nursing Staff Member shall have the ability
     to specify which classes in the system are required for any Program of
     Study in the system";

  Event eChangeClassesOfAProgramOfStudy begin
    occurrence is deferred;
    °Class {}      classes
    : Guarantee g1 is when occurrence ensure me in system.classes;
    ProgramOfStudy program
    : Guarantee g2 is when occurrence ensure me in system.programs;
  end eChangeClassesOfAProgramOfStudy;

  rr5 is
    for all x of PA union NSM
    after eChangeClassesOfAProgramOfStudy within shortDelay
    assert x can achieve bop.program.classes becomes bop.classes;
end system.R5;
```

```
system.R6 begin
  Requirement ~r6
    "The system shall be able to display a report of needed classes for a
     given quarter for all cohorts of all programs for
     Program Administrators/Nursing Staff Members planning purposes"
  : concretisation is rr6;

  Requirement rr6 is
    "The system shall be able to display a report containing, for a
     given quarter, the clinical classes required of each cohort in the
     system that follows a program of study of the Nursing Department";

  Event @eRequestOfClassesNeededByNursingCohortsForQuarter begin
    occurrence is deferred;
    Quarter quarter;
  end eRequest;

  Information
    infoOnClassesNeededByCohortForQuarter (Cohort cohort, Quarter quarter)
    is deferred;

  rr6 is
    after eRequestOfClassesNeededByNursingCohortsForQuarter within shortDelay
    achieve eReport such that contents = {for all x of system.nursingCohorts :
      infoOnClassesNeededByCohortForQuarter (x, bop.quarter)};
end system.R6;
```

```
system.R7 begin
  Requirement ~r7
    "Program Administrators and Nursing Staff Members shall be able to add a
     new clinical site into the system"
  : concretisation is rr7;

  Requirement rr7 is
    "Each Program Administrator or Nursing Staff Member shall be able to
     add new clinical sites into the system";

  Event eAddClinicalSite begin
    occurrence is deferred;
    ClinicalSite site;
  end eAddClinicalSite;

  rr7 is
    for all x of PA union NSM
    after eAddClinicalSite within shortDelay
    assert x can achieve bop.site in system.clinicalSites;
end system.R7;
```

```
system.R8 begin
  Requirement ~r8
    "Program Administrators and Nursing Staff Members shall be able to add a new
     clinical lab section for an existing clinical class into the System"
  : concretisation is {rr8a, rr8b};
  // ..............................................................................
  Requirement rr8a is
    "Each Program Administrator or Nursing Staff Member shall be able
     to add new clinical lab sections to the system";

  Event eAddClinicalLabSection begin
    occurrence is deferred,
    ClinicalLabSection {} clinicalLabSections;
  end eAddClinicalLabSection;

  rr8a is
    for all x of PA union NSM
    after eAddClinicalLabSection within shortDelay
    assert x can achieve bop.clinicalLabSections in system.clinicalLabSections;
  // ..............................................................................
  Requirement rr8b is
    "Each Program Administrator or Nursing Staff Member shall be able
     to add any clinical lab sections in the system to any clinical class
     in the system"

  Event eAddClinicalLabSectionToClinicalClass begin
                       occurrence is deferred,
    ClinicalLabSection {} clinicalLabSections
    : Guarantee g1 is when occurrence ensure me in system.clinicalLabSections;
    ClinicalClass      clinicalClass
    : Guarantee g2 is when occurrence ensure me in system.clinicalClasses;
  end eAddClinicalLabSectionToClinicalClass;

  rr8b is
    for all x of PA union NSM
    after eAddClinicalLabSectionToClinicalClass within shortDelay
    assert x can achieve bop.clinicalLabSections in bop.clinicalClass.sections;
end system.R8;
```

```
system.R9 begin
  Requirement ~r9
    "Program Administrators and Nursing Staff Members shall be able to add
     a student who has registered for a clinical class to a clinical lab
     section for that class"
  : concretisation is rr9;

  Requirement rr9 is
    "Each Program Administrator or Nursing Staff Member shall be able
     to add students registered for a clinical class in the system to
     any clinical lab section in the system for that class";

  Event eAddStudentToClinicalLabSection begin
    occurrence is deferred,
    Student            student
    : Guarantee g1 is when occurrence ensure me in system.students;
    ClinicalLabSection clinicalLabSection
    : Guarantee g2 is when occurrence ensure me in system.sections;
    Guarantee g3 is
      when occurrence
      for some x in system.clinicalclasses
      ensure student in x.students and clinicalLabSection in x.sections;
  end eAddStudentToClinicalLabSection;

  rr9 is
    for all x of PA union NSM
    after eAddStudentToClinicalLabSection within shortDelay
    assert x can achieve bop.student in bop.clinicalLabSection.students;
end system.R9;
```

```
system.R10 begin
  Requirement ~r10
    "The system shall allow a Program Administrator or Nursing staff member
     to remove a student from a clinical lab section"
  : concretisation is rr10;

  Requirement rr10 is
    "Each Program Administrator or Nursing Staff Member shall be able to
     remove any student from any clinical lab section in the system";

  Event eRemoveStudentFromClinicalLabSection begin
    occurrence is deferred,
    Student            student;
    ClinicalLabSection clinicalLabSection
    : Guarantee g1 is when occurrence ensure me in system.clinicalLabSections;
  end eRemoveStudentFromClinicalLabSection;

  rr10 is
    for all x of PA union NSM
    after eRemoveStudentFromClinicalLabSection within shortDelay
    assert x can achieve not bop.student in bop.clinicalLabSection.students;
end system.R10;
```

```
system.R11 begin
  Requirement ~r11
    "The system shall allow a Program Administrator/Nursing Staff Member to move
     a student from one clinical lab section to another clinical lab section
     corresponding to the same clinical class"
  : concretisation is rr11;

  Requirement rr11 is
    "Each Program Administrator or Nursing Staff Member shall be able to move
     any student in any clinical lab section to any another clinical lab
     section corresponding to the same clinical class";

  Event @eMoveStudentToAnotherClinicalLabSection begin
    occurrence is deferred,
    Student            student
    : Guarantee g1 is when occurrence ensure me in system.students;
    ClinicalLabSection clinicalLabSection
    : Guarantee g2 is when occurrence ensure me in system.clinicalLabSections;
    ClinicalLabSection otherClinicalLabSection
    : Guarantee g3 is when occurrence ensure me in system.clinicalLabSections;
    Guarantee g4 is
      when occurrence
      ensure student in clinicalLabSection.students;
    Guarantee g5 is
      when occurrence
      ensure clinicalLabSection.class = otherClinicalLabSection.class;
  end eMoveStudentToAnotherClinicalLabSection;

  rr11 is
    for all x of PA union NSM
    after eMoveStudentToAnotherClinicalLabSection within shortDelay
    assert x can achieve not bop.student in bop.clinicalLabSection.     students
                         and bop.student in bop.otherClinicalLabSection.students;
end system.R11;
```

```
system.R12 begin
  Requirement ~r12
    "Program Administrators/Nursing Staff Members shall be able to cancel
     a clinical lab section only if there are no students registered for that
     clinical lab section"
  : concretisation is rr12;

  Requirement rr12 is
    "Each Program Administrator or Nursing Staff Member shall be able to cancel
     any clinical lab section in the system that has no registered students";

  Event @eCancelClinicalLabSection begin
    occurrence is deferred,
    ClinicalLabSection clinicalLabSection;
    Guarantee g1 is
      when occurrence
      ensure clinicalLabSection in system.clinicalLabSections;
    Guarantee g2 is
      when occurrence
      ensure clinicalLabSection.students = {};
  end eCancelClinicalLabSection;

  rr12 is
    for all x of PA union NSM
    after eCancelClinicalLabSection within shortDelay
    assert x can achieve not bop.clinicalLabSection in system.clinicalLabSections;
end system.R12;
```

```
system.R13 begin
  Requirement ~r13
    "A Program Administrator/Nursing Staff Member shall be able to add a new
     nursing student to the system"
  : concretisation is rr13;

  Requirement rr13 is
    "Each Program Administrator or Nursing Staff Member shall be able to add
     a new nursing student to the system";

  Event @eAddNewStudent begin
    occurrence is deferred,
    Student student;
  end eAddNewStudent;

  rr13 is
    for all x of PA union NSM
    after eAddNewStudent within shortDelay
    assert x can achieve bop.student in system.nursingStudents;
end system.R13;
```

```
system.R14 begin
  Requirement ~r14
    "A staff member shall have the ability to manage the progress of a part
     time student, which includes modifying their sequence of classes and
     timeline of classes"
  begin
    concretisation is deferred;
    Guard !g14 is ensure rr14 in concretisation;
  end r21;

  Requirement rr14
    "Each Nursing Staff Member shall be able to modify the sequence of classes
     of any part time student in the system";

  Event @eManageStudentSequenceOfClasses begin
    occurrence is deferred,
    Student           student begin
      Guarantee g1 is when occurrence ensure me in system.students;
      Guarantee g2 is when occurrence ensure me.partTime;
    end student;
    SequenceOfClasses sequenceOfClasses
    : Guarantee g3 is when occurrence ensure me in system.sequencesOfClasses;
  end eManageStudentSequenceOfClasses;

  rr14 is
    for all x of NSM
    after eManageStudentSequenceOfClasses within shortDelay
    assert x can achieve
      bop.student.sequenceOfClasses becomes bop.sequenceOfClasses;
end system.R14;
```

```
system.R15 begin
  Requirement ~r15
    "The system shall be able to display a printable summary for
     individual cohorts, which will include the students enlisted,
     the Program of study, sequence of classes, cohort progress
     through the program, and timeline of completion"
  : concretisation is rr15;

  Requirement ~rr15
    "The  system shall be able to display a printable summary for
     any cohort in the system"
    "That summary shall include its enlisted students"
    "That summary shall include its program of study"
    "That summary shall include its sequence of classes"
    "That summary shall include its progress through the program
     of study"
    "That summary shall include its timeline of completion";

  Event @eRequestForCohortSummary begin
    occurrence is deferred;
    Cohort cohort
    : Guarantee g1 is when occurrence ensure me in system.cohorts;
  end eRequestForCohortSummary;

  Information infoOnStudentsEnlistedInCohort     (Cohort cohort)
    is deferred;
  Information infoOnProgramOfStudyOfCohort       (Cohort cohort)
    is deferred;
  Information infoOnSequenceOfClassesOfCohort    (Cohort cohort)
    is deferred;
  Information infoOnProgressOfCohort             (Cohort cohort)
    is deferred;
  Information infoOnTimelineOfCompletionOfCohort (Cohort cohort)
    is deferred;

  EReport eReport;

  rr15 is
    after eRequestForCohortSummary
    achieve eReport such that
    { infoOnStudentsEnlistedInCohort     (bop.cohort),
      infoOnProgramOfStudyOfCohort       (bop.cohort),
      infoOnSequenceOfClassesOfCohort    (bop.cohort),
      infoOnProgressOfCohort             (bop.cohort),
      infoOnTimelineOfCompletionOfCohort (bop.cohort)
    } in eReport.contents;
end system.R15;
```

```
system.R16 begin
  Requirement ~r16
    "The system shall be able to display a printable summary for individual
     clinical sites  which will include information on labs being held at
     that site for the given quarter (Clinical site, general information on
     labs held at that location, dates and times of all labs held there)"
  : concretisation is rr16;

  Requirement rr16
    "The system shall be able to display a printable summary for any
     clinical site in the system"
    "That summary shall include information on each lab being held at
     that site for each quarter";      // Whatever that means

  Event @eRequestForClinicalSiteSummary begin
                 occurrence is deferred;
    ClinicalSite clinicalSite;
    : Guarantee g1 is when occurrence ensure me in system.clinicalSites;
  end eRequestForClinicalSiteSummary;

  Information infoOnLabsAtClinicalSite (ClinicalSite clinicalSite, Quarter quarter)
    is deferred;

  rr16 is
    after eRequestForClinicalSiteSummary within shortDelay
    for all quarter of Quarter
    achieve eReport such that
      infoOnLabsAtClinicalSite (bop.clinicalSite, quarter) in contents;
end system.R16;
```

```
system.R17 begin
  Requirement ~r17
    "The system shall be able to display a printable summary for individual
     clinical labs, which will include (but not be limited to) class name,
     class number, lab section, listing of students enrolled, instructor,
     day(s), time and location"
  : concretisation is rr17;

  Requirement rr17
    "The system shall be able to display a printable summary for any
     clinical lab section in the system"
    "That summary shall include class name"
    "That summary shall include class ID"
    "That summary shall include lab section"    // Whatever that means
    "That summary shall include a listing of students enrolled"
    "That summary shall include information on the instructor"
    "That summary shall include information on days(s) and time"
    "That summary shall include information on location";

  Event @eRequestForClinicalLabSectionSummary begin
    occurrence is deferred;
    ClinicalLabSection clinicalLabSection
    : Guarantee g1 is when occurrence ensure me in system.clinicalLabSections;
  end eRequestForClinicalLabSectionSummary;

  Information infoOnClinicalLabSectionClassName
    (ClinicalLabSection clinicalLabSection) is deferred;
  Information infoOnClinicalLabSectionClassID
    (ClinicalLabSection clinicalLabSection) is deferred;
  Information infoOnClinicalLabSectionLabSection
    (ClinicalLabSection clinicalLabSection) is deferred;
  Information infoOnClinicalLabSectionStudents
    (ClinicalLabSection clinicalLabSection) is deferred;
  Information infoOnClinicalLabSectionInstructor
    (ClinicalLabSection clinicalLabSection) is deferred;
  Information infoOnClinicalLabSectionDaysAndTime
    (ClinicalLabSection clinicalLabSection) is deferred;
  Information infoOnClinicalLabSectionLocation
    (ClinicalLabSection clinicalLabSection) is deferred;

  rr17 is
    after eRequestForClinicalLabSectionSummary within shortDelay
    achieve eReport such that
          infoOnClinicalLabSectionClassName   (bop.clinicalLabSection) in contents
      and infoOnClinicalLabSectionClassID     (bop.clinicalLabSection) in contents
      and infoOnClinicalLabSectionLabSection  (bop.clinicalLabSection) in contents
      and infoOnClinicalLabSectionStudents    (bop.clinicalLabSection) in contents
      and infoOnClinicalLabSectionInstructor  (bop.clinicalLabSection) in contents
      and infoOnClinicalLabSectionDaysAndTime (bop.clinicalLabSection) in contents
      and infoOnClinicalLabSectionLocation    (bop.clinicalLabSection) in contents;
end system.R17;
```

```
system.R18 begin
  Requirement ~r18
    "The system shall be able to display a printable summary for individual
     nursing students, which will include (but not be limited to) student name,
     student ID, admission date, classes, credits, GPA, and the cohort that
     the student is enrolled in"
  : concretisation is rr18;

  Requirement rr18
    "The system shall be able to display a printable summary for any
     nursing student in the system"
    "That summary shall include the name           of the student"
    "That summary shall include the Id             of the student"
    "That summary shall include the admission date of the student "
    "That summary shall include the classes        of the student"
    "That summary shall include the credits        of the student"
    "That summary shall include the GPA            of the student"
    "That summary shall include the cohort         of the student";

  Event @eRequestForNursingStudentSummary begin
    occurrence is deferred;
    Student student
    : Guarantee g1 is when occurrence ensure me in system.nursingStudents;
  end eRequestForNursingStudentSummary;

  Information infoOnStudentName          (Student student) is deferred;
  Information infoOnStudentId            (Student student) is deferred;
  Information infoOnStudentAdmissionDate (Student student) is deferred;
  Information infoOnStudentClasses       (Student student) is deferred;
  Information infoOnStudentCredits       (Student student) is deferred;
  Information infoOnStudentGPA           (Student student) is deferred;
  Information infoOnStudentCohort        (Student student) is deferred;

  rr18 is
    after eRequestForNursingStudentSummary within shortDelay
    achieve eReport such that
          infoOnStudentName          (bop.student) in contents
      and infoOnStudentId            (bop.student) in contents
      and infoOnStudentAdmissionDate (bop.student) in contents
      and infoOnStudentClasses       (bop.student) in contents
      and infoOnStudentCredits       (bop.student) in contents
      and infoOnStudentGPA           (bop.student) in contents
      and infoOnStudentCohort        (bop.student) in contents;
end system.R18;
```

```
system.R19 begin
  Requirement r19
    "The system shall be able to display a printable summary of the system log";

  Event @eRequestForSystemLogSummary is deferred;

  Information infoOnSystemLog is deferred;  // Whatever that means

  r19 is
    after eRequestForSystemLogSummary within shortDelay
    achieve eReport such that contents = infoOnSystemLog;
end system.R19;
```

```
system.R20 begin
  Requirement ~r20
    "The system will notify affected parties when changes occur
     affecting cohorts, including but not limited to changes to the
     sequence for a cohort's program of study and changes to a given
     week's schedule (lab cancelled this week due to instructor
     illness)"
  begin
    concretisation is deferred;
    Guard !g20 is ensure {rr20a, rr20b} in concretisation;
  end r21;

  Cohort: ENotification eProgramOfStudyChanged is deferred;
  Cohort: ENotification eScheduleChanged       is deferred;

  Requirement rr20a
    "The system shall notify all persons in the system affected by
     any change in the program of study for any cohort in the
     system"
    is for all x of system.cohorts
       after x.programOfStudy mutates
       achieve eProgramOfStudyChanged;

  Requirement rr20b
    "The system shall notify all persons in the system affected by
     any change in the schedule for any cohort in the system"
    is for all x of system.cohorts
       after OR (x.preferredSequence.sections.beginning MUTATES) or
             OR (x.preferredSequence.sections.°duration MUTATES)
       achieve x.eScheduleChanged;
end system.R20;
```

```
system.R21 begin
  Requirement ~r21
    "The system will notify affected parties when changes occur affecting
     specific students, including but not limited to changing a student's
     status from full time to part time, and moving a student to a different
     clinical section"
  begin
    concretisation is deferred;
    Guard !g21 is ensure {rr21a, rr21b} in concretisation;
  end r21;
  // .........................................................................
  Requirement rr21a
    "The system shall notify all persons affected by any change in
     the full time / part time status of any student in the system"
    is for all x of system.students
       when x.partTime changes within reasonableDelay
       achieve x.eNotification;
  // .........................................................................
  Requirement rr21b
    "The system shall notify all persons affected by the moving of any student
     in the system to a different clinical section in the system" is
    is for all x of system.students
       when x.clinicalLabSection changes within reasonableDelay
       achieve x.eNotification;
end system.R21;
```

```
system.R22 begin
  Requirement ~r22
    "Program Administrators/Nursing Staff Members shall have the ability to
     modify information relating to cohorts, including cohort identifier,
     program of study, preferred sequence of classes and quarters that a cohort
     will be taking specific classes"
  begin
    concretisation is deferred;
    Guard !g22 is ensure {rr22a, rr22b, rr22c, rr22d} in concretisation;
  end r22;
  // ..............................................................................
  Requirement rr22a
    "Each Program Administrator or Nursing Staff Member shall have the
    ability to modify the identifier of any cohort in the system";

  Event eChangeCohortIdentifier begin
    Cohort  cohort
    : Guarantee g1 is when occurrence ensure me in system.cohorts;
    Id      newId;
  end eChangeCohortIdentifier;

  rr22a is
    for all x of PA union NSM
    after eChangeCohortIdentifier within shortDelay
    assert x can achieve bop.cohort.id becomes bop.newId;
  // ..............................................................................
  Requirement rr22b
    "Each Program Administrator or Nursing Staff Member shall have the ability to
     modify the program of study of any cohort in the system";

  Event eChangeCohortProgramOfStudy begin
    Cohort            cohort
    : Guarantee g1 is when occurrence ensure me in system.cohorts;
    ProgramOfStudy    newProgramOfStudy
    : Guarantee g2 is when occurrence ensure me in system.programs;
  end eChangeCohortProgramOfStudy;

  rr22b is
    for all x of PA union NSM
    after eChangeCohortProgramOfStudy within shortDelay
    assert x can achieve bop.cohort.programOfStudy becomes bop.newProgramOfStudy;
  // ..............................................................................
  Requirement rr22c
    "Each Program Administrator or Nursing Staff Member shall have the ability to
     modify the preferred sequence of classes of any cohort in the system";

  Event eChangeCohortPreferredSequence begin
    Cohort            cohort
    : Guarantee g1 is when occurrence ensure me in system.cohorts;
    SequenceOfClasses newPreferredSequence;
  end eChangeCohortPreferredSequence;

  rr22c is
    for all x of PA union NSM
    after eChangeCohortPreferredSequence within shortDelay
    assert x can achieve
      bop.cohort.preferredSequence becomes bop.newPreferredSequence;
  // ................................................................................
  Requirement rr22d
    "Each Program Administrator or Nursing Staff Member shall have the ability to
     modify the quarters that any cohort in the system will be taking specific
     classes";

  Event eChangeCohortClassesQuarters begin
    Cohort            cohort
    : Guarantee g1 is when occurrence ensure me in system.cohorts;
    SequenceOfClasses newSequenceOfClasses;
  end eChangeCohortClassesQuarters;

  rr22d is
    for all x of PA union NSM
    after eChangeCohortClassesQuarters within shortDelay
    assert x can achieve
      bop.cohort.preferredSequence becomes bop.newPreferredSequence;
end system.R22;
```

```
system.R23 begin
  Requirement ~r23
    "Program Administrators/Nursing Staff Members shall have the ability
     to modify information relating to a Program of Study within
     the Nursing Department, including the Program of study name, and
     required classes for that Program of Study"
  begin
    concretisation is deferred;
    Guard !g23 is ensure {rr23a, rr23b} in concretisation;
  end r23;
  // ..............................................................................
  Requirement rr23a is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the name of any Program of study within the
     Nursing Department";

  Event eChangeProgramOfStudyName begin
    ProgramOfStudy  programOfStudy
    : Guarantee g1 is when occurrence ensure me in system.nursingPrograms;
    Name            newName;
  end eChangeProgramOfStudyName;

  rr23a is
    for all x of PA union NSM
    after eChangeProgramOfStudyName within shortDelay
    assert x can achieve bop.programOfStudy.name becomes bop.newName;
  // ..............................................................................
  Requirement rr23b is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the classes required for any Program of study
     within the Nursing Department";

  Event eChangeProgramOfStudyRequiredClasses begin
    ProgramOfStudy programOfStudy
    : Guarantee g1 is when occurrence ensure me in system.nursingPrograms;
    °Class {}      newRequiredClasses
    : Guarantee g2 is when occurrence ensure me in system.classes;
  end eChangeProgramOfStudyRequiredClasses;

  rr23b is
    for all x of PA union NSM
    after eChangeProgramOfStudyRequiredClasses within shortDelay
    assert x can achieve
      bop.programOfStudy.requiredClasses becomes bop.newRequiredClasses;
end system.R23;
```

```
system.R24 begin
  Requirement ~r24
    "Program Administrators/Nursing Staff Members shall have the ability
     to modify information relating to a Clinical Site, including the clinical
     site name, site contact person, contact information, and address"
  begin
    concretisation is deferred;
    Guard !g24 is ensure {rr24a, rr24b, rr24c, rr24d} in concretisation;
  end r24;
  // ..................................................................
  Requirement rr24a is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the name of any clinical site in the system";

  Event eChangeClinicalSiteName begin
    ClinicalSite  clinicalSite
    : Guarantee g1 is when occurrence ensure me in system.clinicalSites;
    Name          newName;
  end eChangeClinicalSiteName;

  rr24a is
    for all x of PA union NSM
    after eChangeClinicalSiteName within shortDelay
    assert x can achieve bop.clinicalSite.name becomes bop.newName;
  // ..................................................................
  Requirement rr24b is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the contact person of any clinical site in the system";

  Event eChangeClinicalSiteContactPerson begin
    ClinicalSite  clinicalSite
    : Guarantee g1 is when occurrence ensure me in system.clinicalSites;
    Person        newContactPerson;
  end eChangeClinicalSiteContactPerson;

  rr24b is
    for all x of PA union NSM
    after eChangeClinicalSiteContactPerson within shortDelay
    assert x can achieve
      bop.clinicalSite.contactPerson becomes bop.newContactPerson;
  // ..................................................................
  Requirement rr24c is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the contact information of any clinical site
     in the system";

  Event eChangeClinicalSiteContactInformation begin
    ClinicalSite  clinicalSite
    : Guarantee g1 is when occurrence ensure me in system.clinicalSites;
    Information   newContactInformation;
  end eChangeClinicalSiteContactInformation;

  rr24c is
    for all x of PA union NSM
    after eChangeClinicalSiteContactInformation within shortDelay
    assert x can achieve
      bop.clinicalSite.contactInformation becomes bop.newContactInformation;
  // ..............................................................................
  Requirement rr24d is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the address of any clinical site in the system";

  Event eChangeClinicalSiteAddress begin
    ClinicalSite  clinicalSite
    : Guarantee g1 is when occurrence ensure me in system.clinicalSites;
    Address       newAddress;
  end eChangeClinicalSiteAddress;

  rr24d is
    for all x of PA union NSM
    after eChangeClinicalSiteAddress within shortDelay
    assert x can achieve bop.clinicalSite.address becomes bop.newAddress;
end system.R24;
```

```
system.R25 begin
  Requirement ~r25
    "Program Administrators/Nursing Staff Members shall have the ability
     to modify information relating to a Clinical Lab Section, including
     the clinical site name, the department/section for the lab, the class
     instructor, day and time of lab, and contact information"
  begin
    concretisation is deferred;
    Guard !g25 is ensure {rr25a, rr25b, rr25c, rr25d, rr25e} in concretisation;
  end r25;
  // ..............................................................................
  Requirement rr25a is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the clinical site name of any Clinical Lab Section
     in the system";

  Event eChangeClinicalLabSectionSite begin
    ClinicalLabSection  clinicalLabSection
    : Guarantee g1 is when occurrence ensure me in system.clinicalLabSections;
    Name                newName;
  end eChangeClinicalLabSectionSite;

  rr25a is
    for all x of PA union NSM
    after eChangeClinicalLabSectionSite within shortDelay
    assert x can achieve
      bop.clinicalLabSection.site.name becomes bop.newName;
  // ..............................................................................
  Requirement rr25b is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the department of any Clinical Lab Section
     in the system";

  Event eChangeClinicalLabSectionDepartment begin
    ClinicalLabSection  clinicalLabSection
    : Guarantee g1 is when occurrence ensure me in system.clinicalLabSections;
    Department          newDepartment;
  end eChangeClinicalLabSectionDepartment;

  rr25b is
    for all x of PA union NSM
    after eChangeClinicalLabSectionDepartment within shortDelay
    assert x can achieve
      bop.clinicalLabSection.department becomes bop.newDepartment;
  // ..............................................................................
  Requirement rr25c is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the class instructor of any Clinical Lab Section
     in the system";

  Event eChangeClinicalLabSectionInstructor begin
    ClinicalLabSection  clinicalLabSection
    : Guarantee g1 is when occurrence ensure me in system.clinicalLabSections;
    Instructor          newInstructor;
  end eChangeClinicalLabSectionInstructor;

  rr25c is
    for all x of PA union NSM
    after eChangeClinicalLabSectionInstructor within shortDelay
    assert x can achieve
      bop.clinicalLabSection.instructor becomes bop.newInstructor;
  // ..............................................................................
  Requirement rr25d is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the day and time of any Clinical Lab Section
     in the system";

  Event eChangeClinicalLabSectionDate begin
    ClinicalLabSection  clinicalLabSection
    : Guarantee g1 is when occurrence ensure me in system.clinicalLabSections;
    Date                newDate;
  end eChangeClinicalLabSectionDate;

  rr25d is
    for all x of PA union NSM
    after eChangeClinicalLabSectionDate within shortDelay
    assert x can achieve
      bop.clinicalLabSection.beginning becomes bop.newDate;
  // ..............................................................................
  Requirement rr25e is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the contact information of any Clinical Lab Section
     in the system";

  Event eChangeClinicalLabSectionContactInformation begin
    ClinicalLabSection  clinicalLabSection
    : Guarantee g1 is when occurrence ensure me in system.clinicalLabSections;
    Information         newContactInformation;
  end eChangeClinicalLabSectionContactInformation;

  rr25e is
    for all x of PA union NSM
    after eChangeClinicalLabSectionContactInformation within shortDelay
    assert x can achieve
      bop.clinicalLabSection.contactInformation becomes bop.newContactInformation;
end system.R25;
```

```
system.R26 begin
  Requirement ~r26
    "Program Administrators/Nursing Staff Members shall have the ability
     to modify information relating to a Nursing Student, including student ID,
     student name, phone number, e-mail, status (full time or part time),
     and program of study"
  begin
    concretisation is deferred;
    Guard !g26 is
      ensure {rr26a, rr26b, rr26c, rr26d, rr26e, rr26f} in concretisation;
  end r26;
  // ..........................................................................
  Requirement rr26a is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the name of any Nursing Student in the system";

  Event eChangeNursingStudentName begin
    Student  student
    : Guarantee g1 is when occurrence ensure me in system.nursingStudents;
    Name     newName;
  end eChangeNursingStudentName;

  rr26a is
    for all x of PA union NSM
    after eChangeNursingStudentName within shortDelay
    assert x can achieve bop.student.name becomes bop.newName;
  // ..........................................................................
  Requirement rr26b is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the ID of any Nursing Student in the system";

  Event eChangeNursingStudentId begin
    Student  student
    : Guarantee g1 is when occurrence ensure me in system.nursingStudents;
    Id       newId;
  end eChangeNursingStudentId;

  rr26b is
    for all x of PA union NSM
    after eChangeNursingStudentId within shortDelay
    assert x can achieve bop.student.id becomes bop.newId;
  // ..........................................................................
  Requirement rr26c is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the phone number of any Nursing Student in the system";

  Event eChangeNursingStudentPhoneNumber begin
    Student     student
    : Guarantee g1 is when occurrence ensure me in system.nursingStudents;
    PhoneNumber newPhoneNumber;
  end eChangeNursingStudentPhoneNumber;

  rr26c is
    for all x of PA union NSM
    after eChangeNursingStudentPhoneNumber within shortDelay
    assert x can achieve bop.student.phoneNumber becomes bop.newPhoneNumber;
  // ..............................................................................
  Requirement rr26d is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the e-mail of any Nursing Student in the system";

  Event eChangeNursingStudentEMail begin
    Student student
    : Guarantee g1 is when occurrence ensure me in system.nursingStudents;
    EMail   newEMail;
  end eChangeNursingStudentEMail;

  rr26d is
    for all x of PA union NSM
    after eChangeNursingStudentEMail within shortDelay
    assert x can achieve bop.student.eMail becomes bop.newEMail;
  // ..............................................................................
  Requirement rr26e is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the status of any Nursing Student in the system";

  Event eChangeNursingStudentStatus begin
    Student student
    : Guarantee g1 is when occurrence ensure me in system.nursingStudents;
    Boolean newStatus;
  end eChangeNursingStudentStatus;

  rr26e is
    for all x of PA union NSM
    after eChangeNursingStudentStatus within shortDelay
    assert x can achieve bop.student.partTime becomes bop.newStatus;
  // ..............................................................................
  Requirement rr26f is
    "Each Program Administrator or Nursing Staff Member shall have the
     ability to modify the program of study of any Nursing Student
     in the system";

  Event eChangeNursingStudentProgramOfStudy begin
    Student        student
    : Guarantee g1 is when occurrence ensure me in system.nursingStudents;
    ProgramOfStudy newProgramOfStudy
    : Guarantee g2 is when occurrence ensure me in system.programs;
  end eChangeNursingStudentProgramOfStudy;

  rr26f is
    for all x of PA union NSM
    after eChangeNursingStudentProgramOfStudy within shortDelay
    assert x can achieve
      bop.student.programOfStudy becomes bop.newProgramOfStudy;
end system.R26;
```