User Requirements Notation Models

for

RaiderNAV

**Table of Contents Placeholder**

**1. Goal Requirements Language**

The GRL models shown in this document demonstrate the high-level goals, as well as the rationales associated with achieving those goals, for the two main actors connected to the RaiderNAV project: Team Getana and Texas Tech students. Diagrams illustrating this information are provided both individually by actor, as well as in an integrated view.

**1.1 Strategies**

Four strategies were evaluated in an effort to determine the best possible framework to satisfy the goals of both Team Getana and Texas Tech students. Team Getana’s primary goals are to get an ‘A’ in the Software Engineering II course and to ensure the RaiderNAV app is popular. The Texas Tech students’ top goals are to find their class locations more easily while still maintaining a high level of privacy regarding their personal information.

Analysis of Team Getana’s goals reveals them to be mutually exclusive to a certain degree. For example, devoting more time to RaiderNAV development is beneficial to the app’s popularity, but detrimental to the group’s grades if the resources necessary are diverted from other important class activities. The case for Texas Tech students is somewhat similar. Utilizing RaiderNAV makes it much easier to locate class locations but does so at the cost of certain private information that must be provided to Google Maps.

Since this is the case, the four strategies evaluated each represent a unique combination of the goals predominantly pursued by the actors.

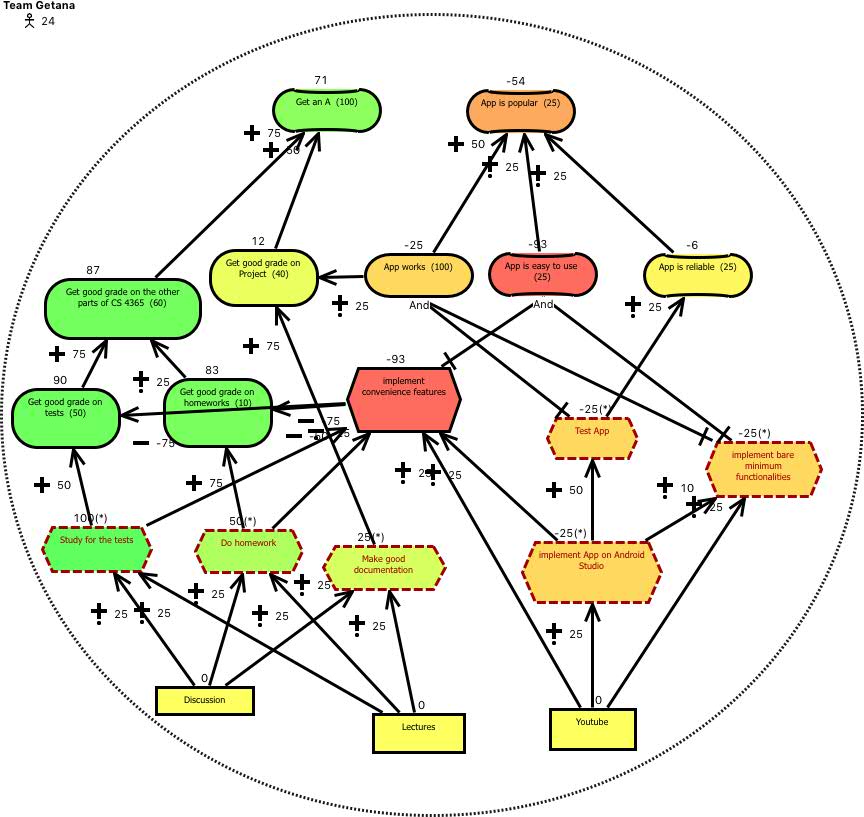
**1.1.1 Combined Strategy #1: Team Getana Deemphasizes Project and Students Avoid RaiderNAV**

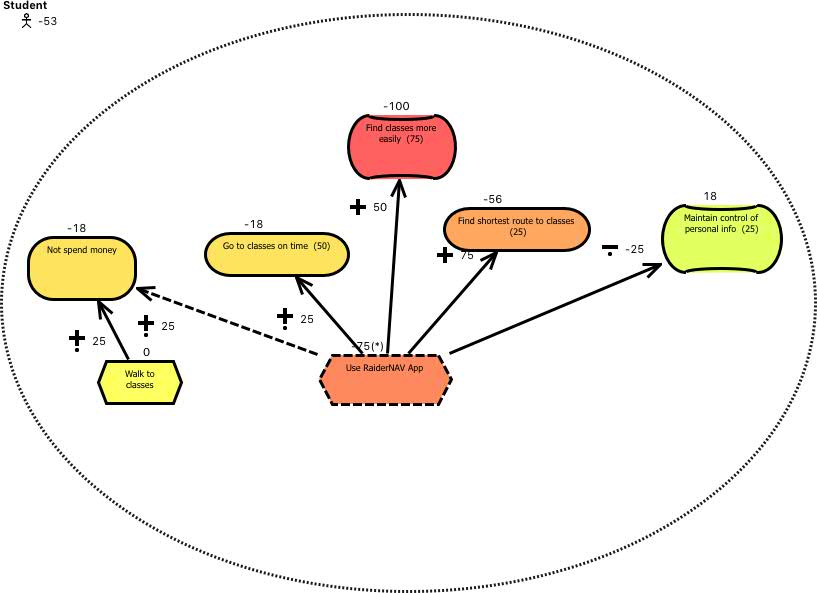
Strategy one represents the case where Team Getana prioritizes tests and homework over the RaiderNAV project and a Texas Tech student elects not to utilize RaiderNAV in order to maintain information privacy. This results in a high satisfaction value for Team Getana’s goal of achieving an ‘A’. However, it also leads to a poorly executed app which is not popular. Meanwhile, the choice to not use the RaiderNAV app effectively preserves the student’s privacy but does not improve their ability to locate classes more easily.

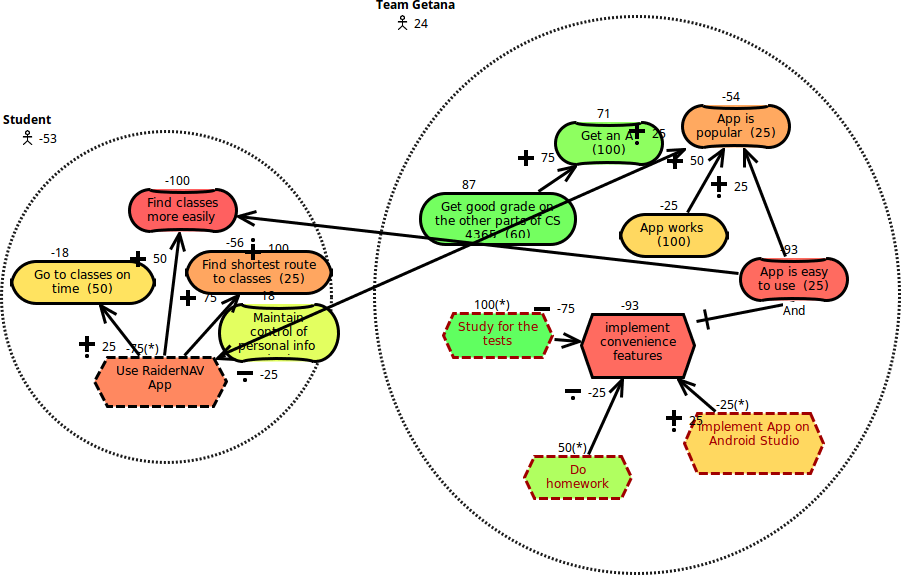
Team Getana model for Strategy #1 is depicted in figure 1.1.1.1

Texas Tech Student model for Strategy #1 is depicted in figure 1.1.1.2

Combined model for Team Getana & Texas Tech Students for Strategy #1 is depicted in figure 1.1.1.3

Figure 1.1.1.1 Team Getana GRL model under strategy 1

Figure 1.1.1.2 Student GRL model under strategy 1

Figure 1.1.1.3 Combined GRL model under strategy 1

**1.1.2 Combined Strategy #2: Team Getana Emphasizes Project and Students Avoid RaiderNAV**

Strategy #2, unlike above, reflects a scenario wherein Team Getana focuses their efforts primarily on the RaiderNAV project, even to the point of neglecting other classwork while a student still chooses not to utilize RaiderNAV. The increased effort put toward the app is reflected in a more well-designed final product that is more popular among users in general, but adversely affects Team Getana’s overall grade in the Software Engineering course. The popularity of the app also makes it slightly easier for an individual student to locate a course even if they choose not to utilize the app, because other students are more readily able to assist, but not meaningfully so in other situations.

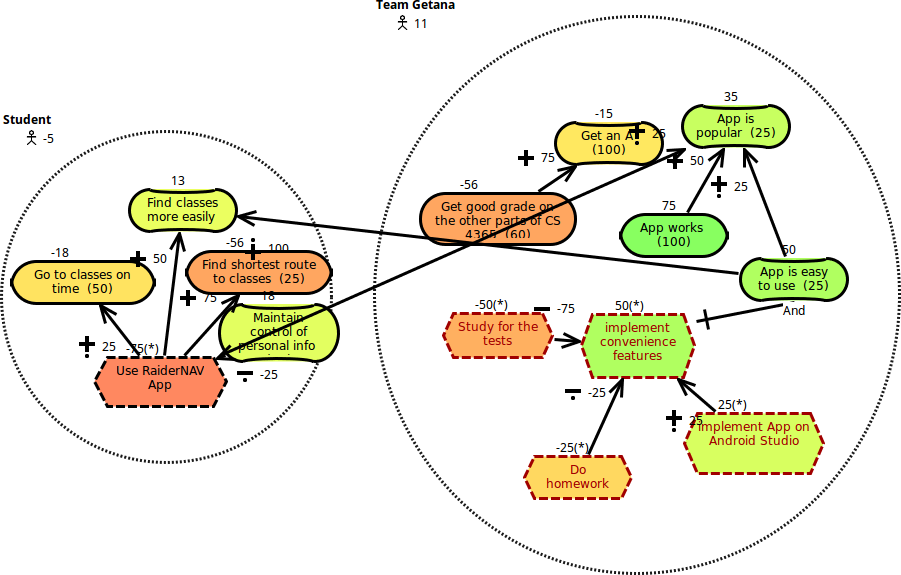
Team Getana model for Strategy #2 is depicted in figure 1.1.2.1

Texas Tech Student model for Strategy #2 is depicted in figure 1.1.2.2

Combined model for Team Getana & Texas Tech Students for Strategy #2 is depicted in figure 1.1.2.3

Figure 1.1.2.1 Team Getana GRL model under strategy 2

Figure 1.1.2.2 Student GRL model under strategy 2

Figure 1.1.2.3 Combined GRL model under strategy 2

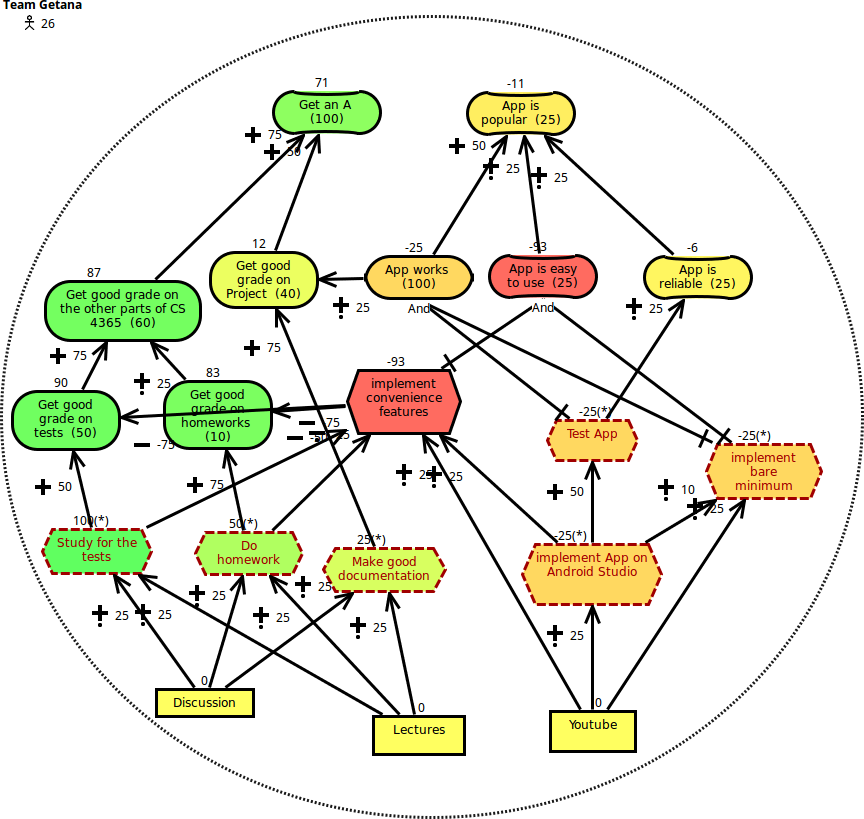
**1.1.3 Combined Strategy #3: Team Getana Deemphasizes Project and Students Use RaiderNAV**

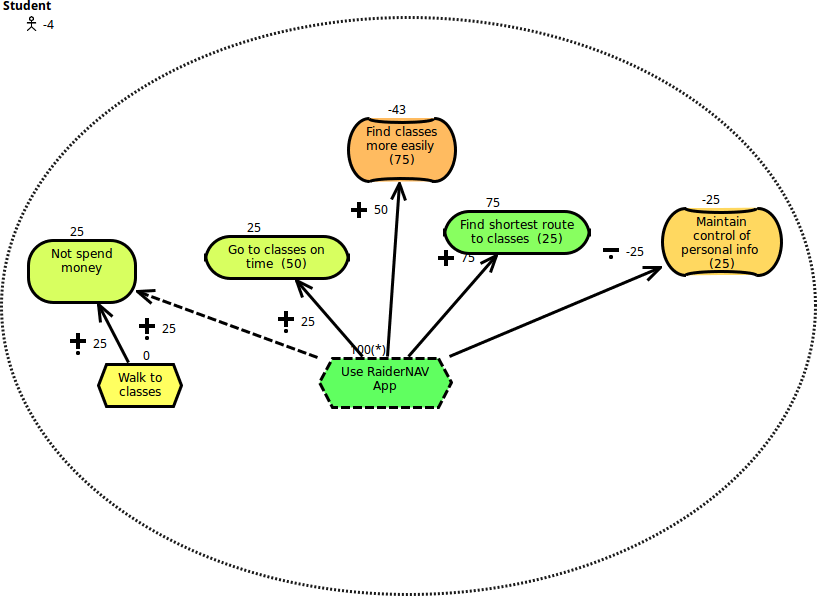
Strategy #3 represents the inverse of strategy #2. In this situation, Team Getana again focuses on homework and tests to improve their grades for Software Engineering II, but the student in question does use RaiderNAV to locate his/her classes. Because Team Getana dedicates more of their resources to other assignments that constitute a larger portion of their overall grade, their chance of getting an ‘A’ significantly improves. Unfortunately, this also results in a decrease in the quality of the RaiderNAV app, which is not as intuitive for the user. Consequently, the student’s ability to more easily find their classes is still improved, although not as much as if the app was better designed.

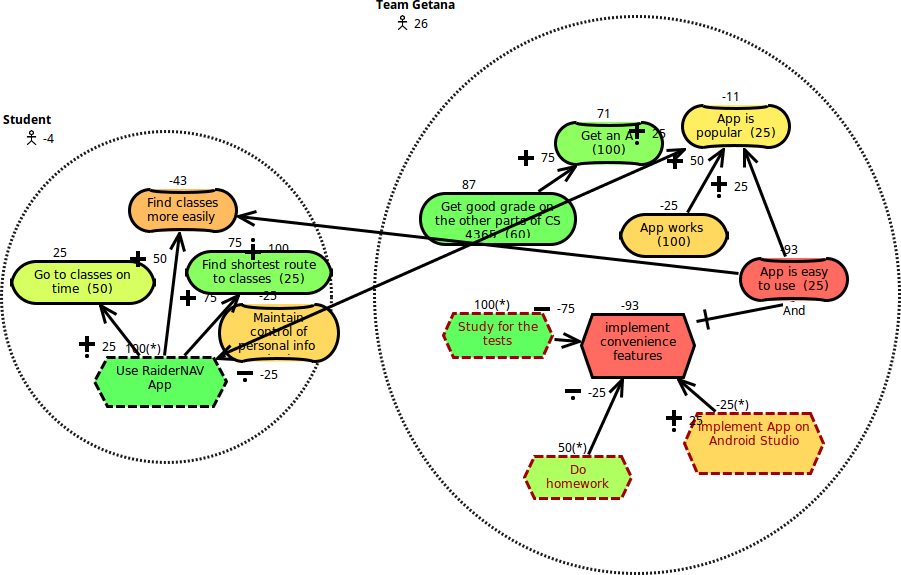
Team Getana model for Strategy #3 is depicted in figure 1.1.3.1

Texas Tech Student model for Strategy #3 is depicted in figure 1.1.3.2

Combined model for Team Getana & Texas Tech Students for Strategy #3 is depicted in figure 1.1.3.3

Figure 1.1.3.1 Team Getana GRL model under strategy 3

Figure 1.1.3.2 Student GRL model under strategy 3

Figure 1.1.3.3 Combined GRL model under strategy 3

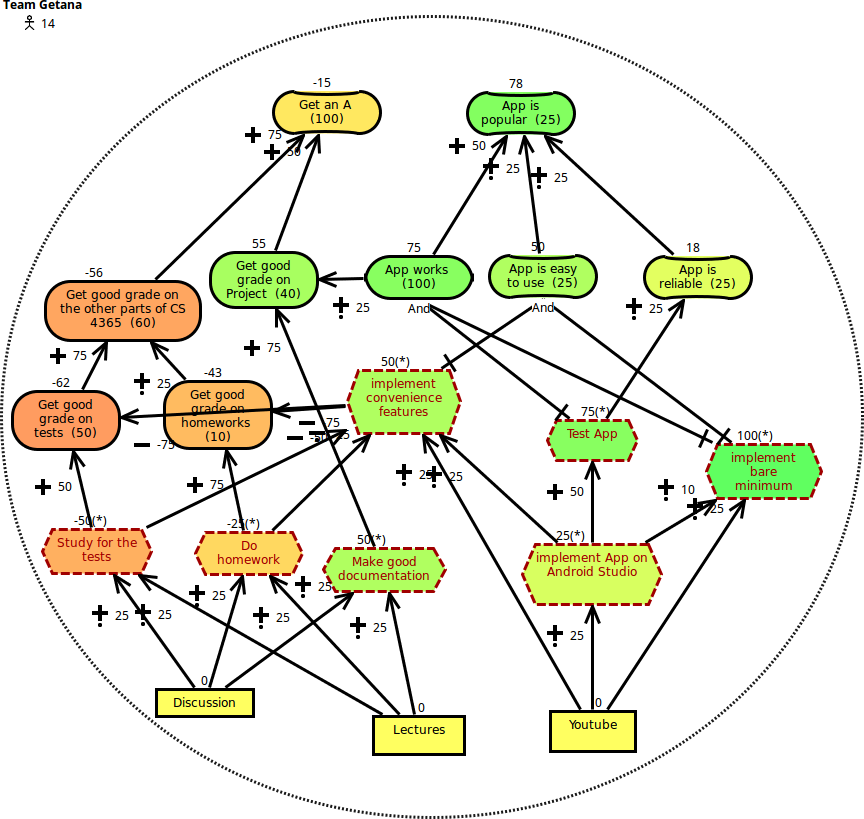
**1.1.4 Combined Strategy #4: Team Getana Emphasizes Project and Students Use RaiderNAV**

The final strategy evaluated, strategy #4, explores the effect when Team Getana prioritizes the development of the RaiderNAV app and a student utilizes the resulting product. As expected the result is a well-produced app that is both easy to use and popular, but at the cost of Team Getana’s overall grade in the Software Engineering II course.

Team Getana model for Strategy #4 is depicted in figure 1.1.4.1

Texas Tech Student model for Strategy #4 is depicted in figure 1.1.4.2

Combined model for Team Getana & Texas Tech Students for Strategy #4 is depicted in figure 1.1.4.3

Figure 1.1.4.1 Team Getana GRL model under strategy 4

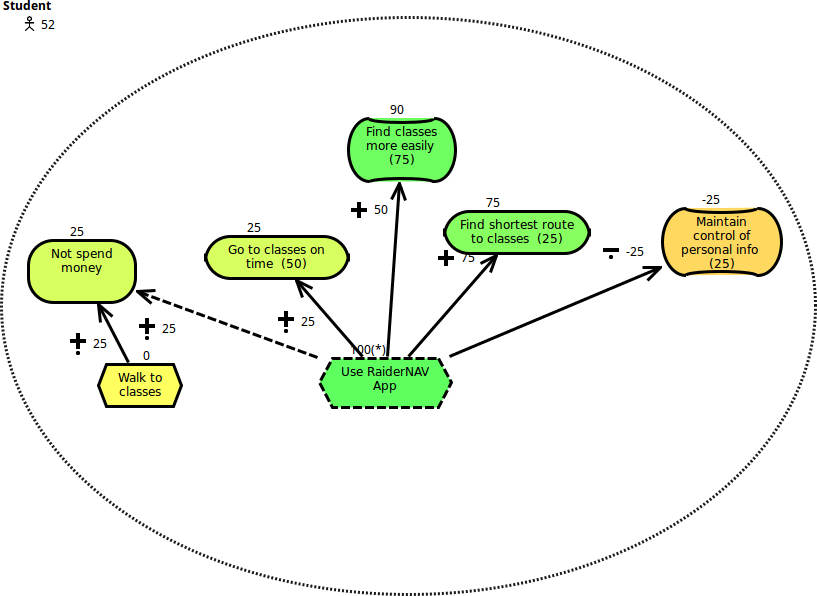


Figure 1.1.4.2 Student GRL model under strategy 4

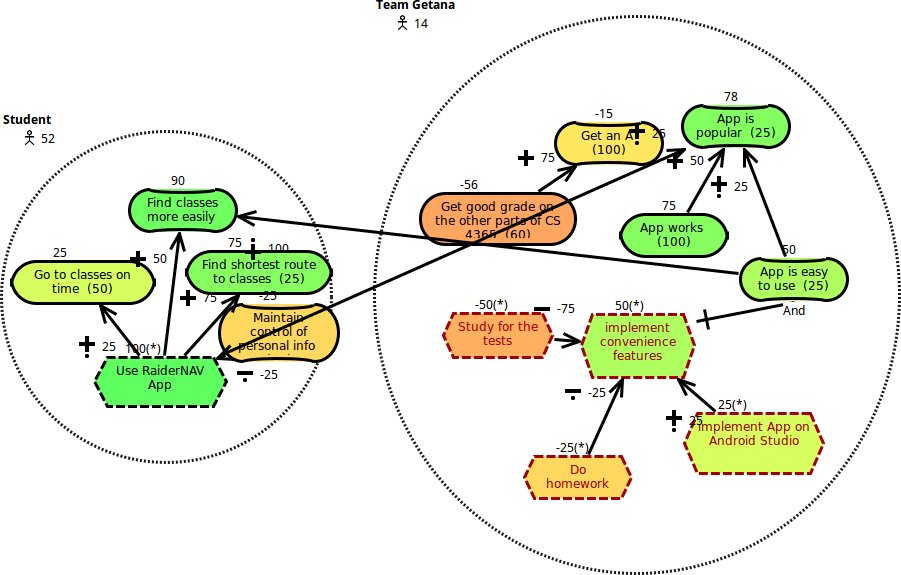


Figure 1.1.4.3 Combined GRL model under strategy 4

**1.2 Analysis of Combined Strategies**

Any analysis of GRL strategies must pay careful consideration to the concerns of each stakeholder or actor. The goals of each party may vary significantly and are often in fact at odds with one another. As such it is beneficial to address each strategy with regard to the actors involved, as well in a cumulative manner.

First, we examine the case of an individual student. The two high-level goals established earlier for a student are to find their class locations more easily and to maintain the privacy of their data. However, these goals are not weighted evenly. In fact, the ability to find class locations easily is shown to be far more important than information privacy. As such, we can expect the strategies that satisfy this goal to produce much higher satisfaction ratings for the student actor. This proves to be the case, and the highest overall satisfaction for the student actor is achieved from strategy 4. Here the student utilizes the RaiderNAV app to improve their navigation skills around campus, and the app itself is well-designed as a result of Team Getana’s heightened efforts towards its development. Conversely, the student’s satisfaction rating is lowest in strategy 1 when he/she does not utilize the RaiderNAV app and Team Getana focuses on other coursework.

The “best” strategy for Team Getana, without regard for Texas Tech students, is markedly different. Again, we see that while Team Getana had two primary high-level goals, to make an ‘A’ in Software Engineering II, and to produce a popular app, one weighs far more heavily than the other. In this case, the goal of making an ‘A’ in the course is more important than producing a popular app. Naturally, strategies 1 and 3, which favor this goal, are better options for Team Getana. Regrettably, these are also the strategies for which the goals of a Texas Tech student are least satisfied.

Decisions regarding the best strategy to implement should strive to satisfy as many of the stakeholders as possible. Achieving such a goal can be expected to involve some degree of compromise and may even leave some stakeholders completely unsatisfied for the collective good. Examining the implementable strategies shown above for the RaiderNAV project demonstrates that only strategy 4 satisfies both concerned actors, if only weakly. Although strategies 1 and 3 produce better results for Team Getana, they do not address the concerns of Texas Tech students. As such, strategy 4 is the best choice for implementation.

**2. Use Case Maps**

Text here explains the layout of the following stuff (i.e., the hierarchical organization of 2.1 and layout of scenarios in 2.2).

# 2.1 Main Sequence Actions

Text here explains that we have 2 (and potentially more later) possible actions on the main screen:

1. Perform an unscheduled navigation
2. Perform an action on a schedule

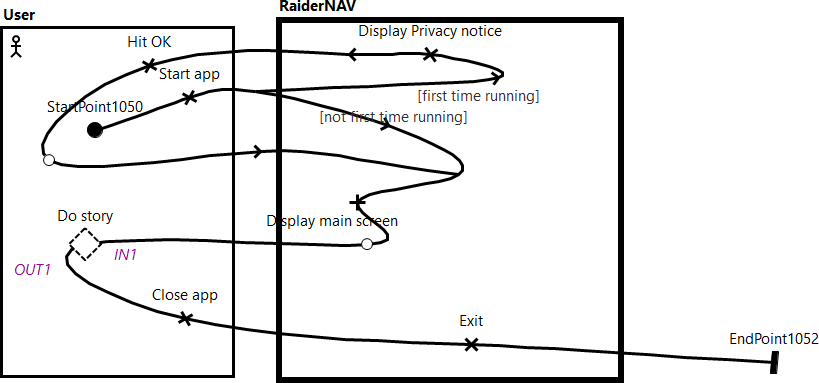


Figure 2.1 Main Sequence UCM

2.1.1 Unscheduled Navigation

Text here explains what happens in an unscheduled navigation.

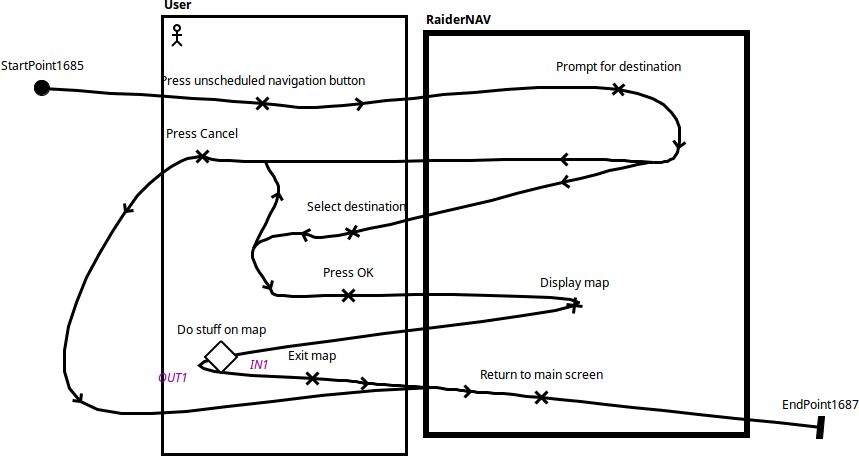


Figure 2.1.1 Unscheduled Navigation UCM

2.1.1.1 Show Shortest Route between Two Locations

When user long-presses two points on the map, the app will show the shortest route between the two locations.

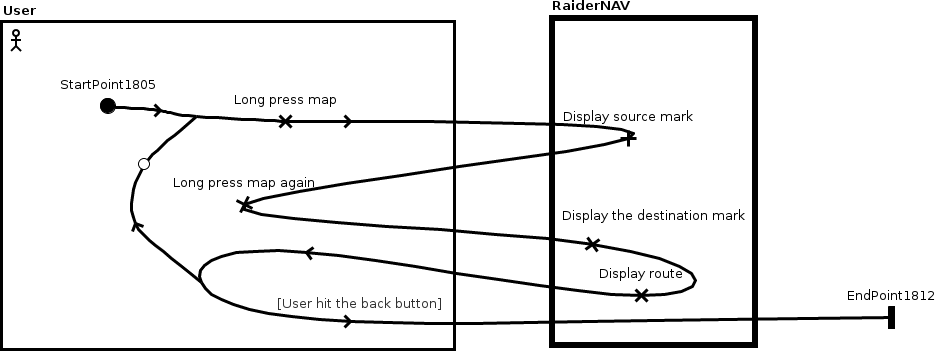


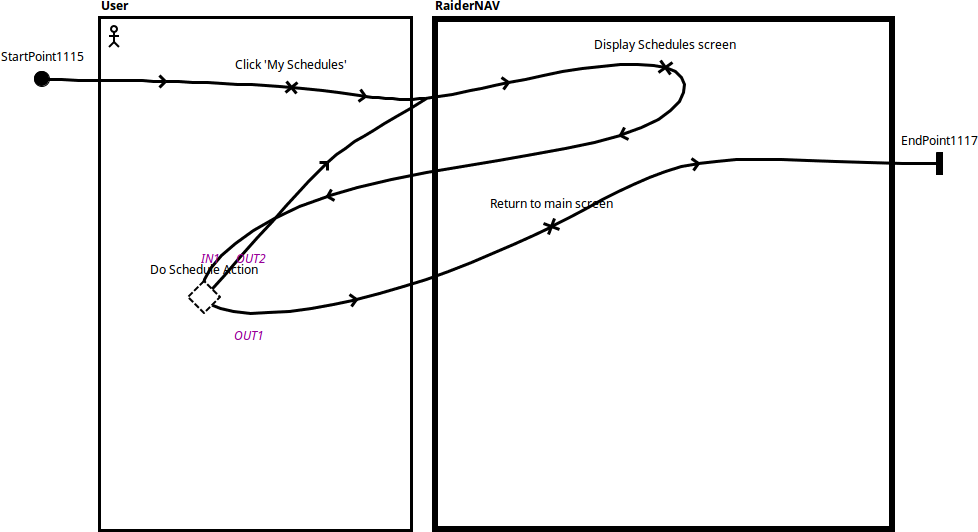
Figure 2.1.1.1 Show the shortest route between two locations

2.1.2 Schedule Action

Text here explains what happens when performing a schedule action.

Text here explains that we have two possible actions from within the main schedules activity:

1. Create a new schedule
2. Perform an action on an existing schedule



Schedule Main Sequence UCM

2.1.2.1 Create New Schedule

Text here explains what happens when creating a new schedule.

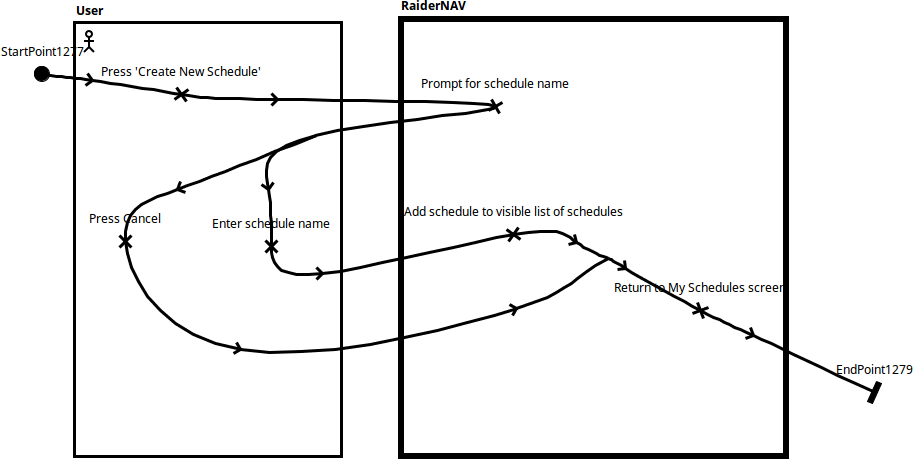


Figure 2.1.2.1 Create New Schedule UCM

2.1.2.2 Edit Existing Schedule

Text here explains what happens when editing an existing schedule.

Text here explains that you can perform 4 actions as part of editing an existing schedule:

1. Create a course
2. Delete the schedule
3. Rename the schedule
4. Edit a course

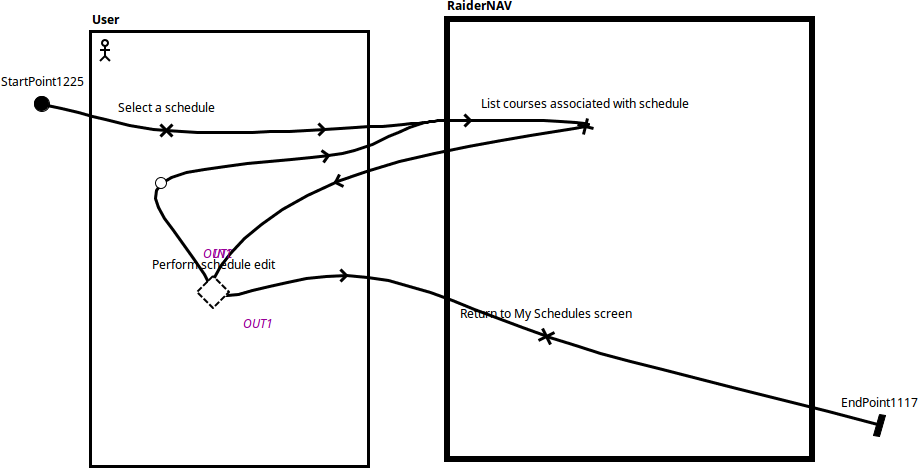


Figure 2.1.2.2 Edit Schedule Main Sequence UCM

2.1.2.2.1 Creating a New Course

Text here explains what happens when creating a new course.

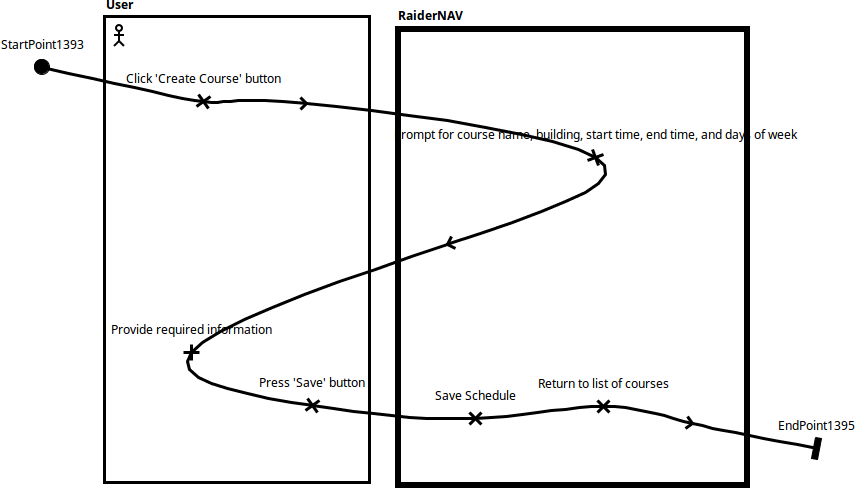
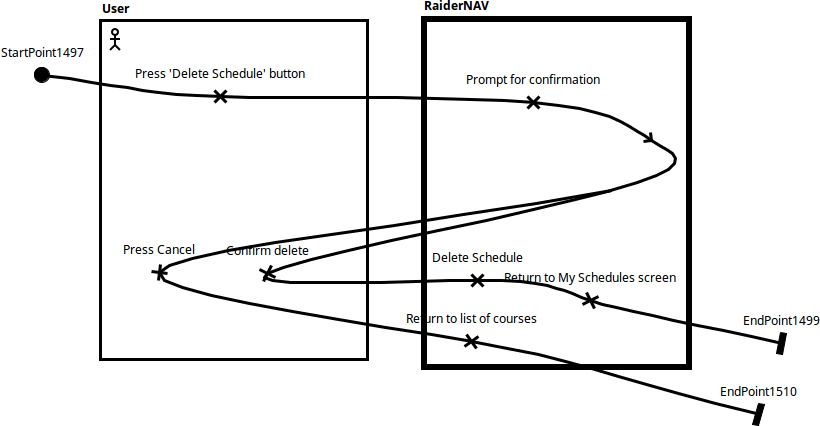


Figure 2.1.2.2.1 Create Course UCM

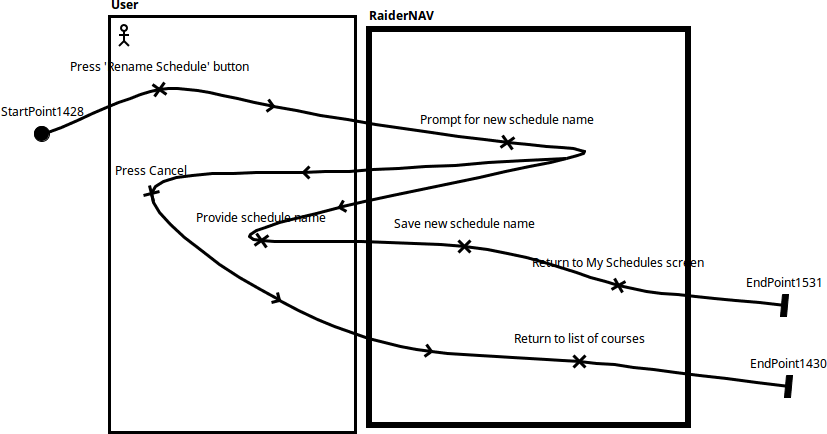
2.1.2.2.2 Deleting a Schedule

Text here explains what happens when deleting a schedule.

Figure 2.1.2.2.2 Delete Schedule UCM

2.1.2.2.3 Renaming a Schedule

Text here explains what happens when renaming a schedule.

Figure 2.1.2.2.3 Rename Schedule UCM

2.1.2.2.4 Editing a Course

Text here explains what happens when editing a course.

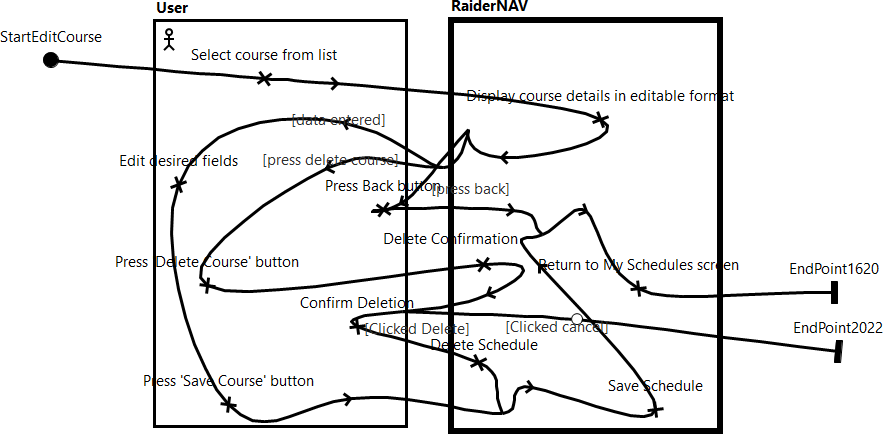
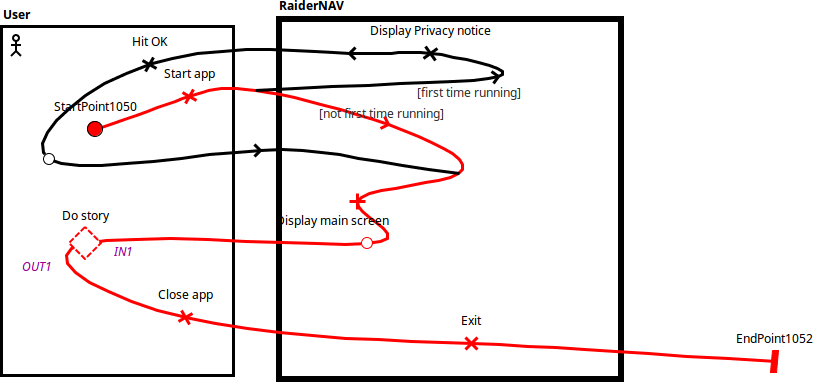


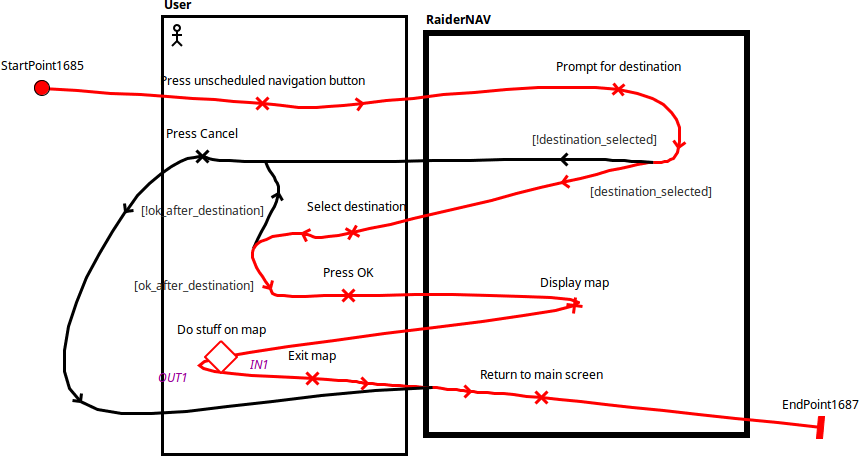
Figure 2.1.2.2.4 Edit Course UCM

2.2 Scenarios

2.2.1 User performs an unscheduled navigation

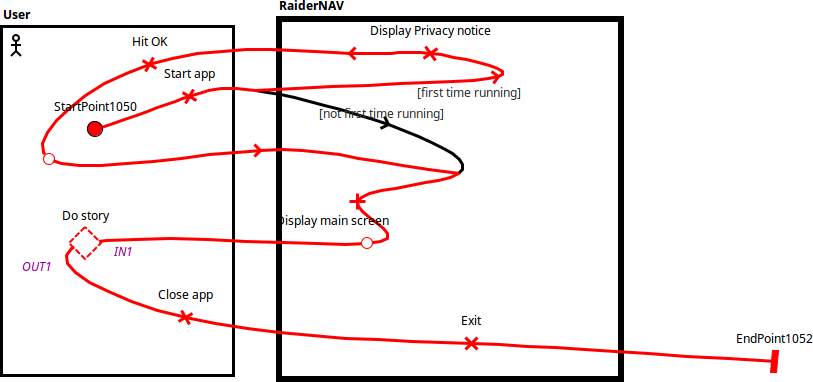
Text here describes what is happening in the diagrams (i.e., the user opens the app, goes through the process of doing an unscheduled navigation, then leaves the map and ultimately closes the app).

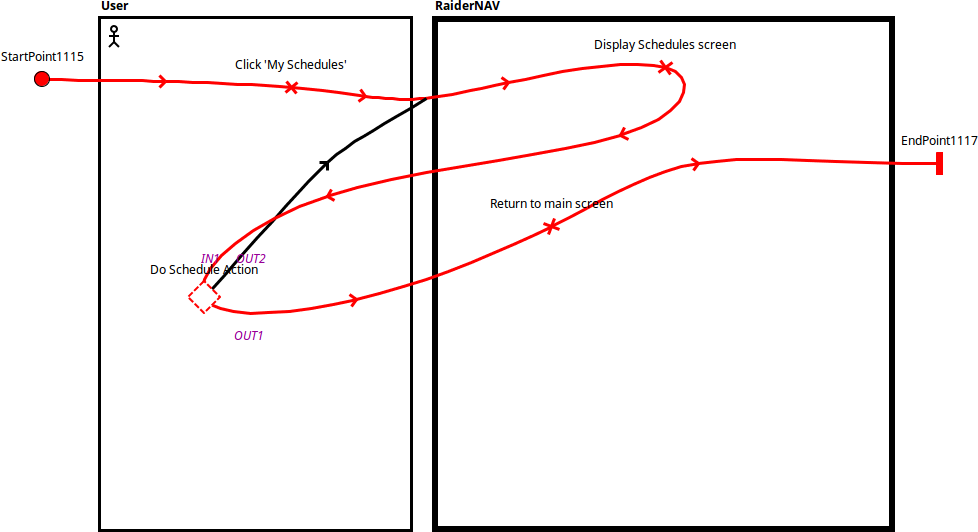
Figure 2.2.1.1 Top Level Sequence

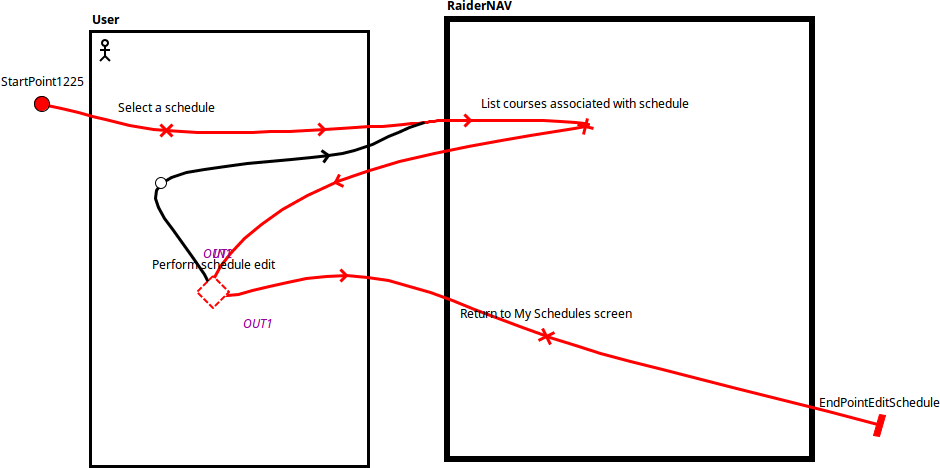
Figure 2.2.1.2 Performing Unscheduled Navigation

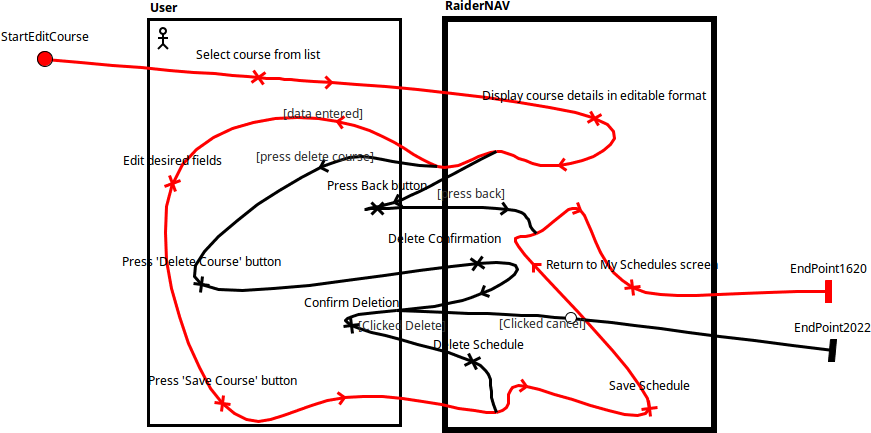
2.2.2 User edits a course in a schedule

Text here describes what is happening in the diagrams (i.e., the user opens the app, goes through the process of editing a course, then leaves the schedule view and ultimately closes the app).

Figure 2.2.2.1 Top Level Sequence

Figure 2.2.2.2 User Enters Schedules View

Figure 2.2.2.3 User Selects a Schedule

Figure 2.2.2.4 User Edits a Course