1 - Predict future numbers of animals:

**Primary Actor:** User

**Stakeholders and Interests:**

- User: Wants an accurate and easy to use application that will show them the effects of certain solutions to the unbalanced nature preserve problem.

- Company: an application which implements the necessary mathematical models which calculate the effects of certain conditions on the current situation and thus showing the best possible solution to the described problem.

**Precondition: -**

**Postcondition:** an exact number of populations per animal will be projected and readable.

**Main Success Scenario (or Basic Flow):**

1. User fills in the required values into the textboxes and presses “calculate”.

2. Application calculates a prediction using the default/selected mathematical model.

3. Application presents the predicted results of future populations.

**Extensions (or Alternative Flows):**

\*a. At any time, System fails:

Assuming System crashes after a calculation had been attempted.

1. User closes the application and opens it again.

2. Application starts up.

3. User attempts the calculation again.

2a. Invalid input:

1. Application sends a pop-up message stating “Please insert a number greater than 0”.
2. User presses “ok” button.
3. Application closes pop-up window.

3a. Switching between mathematical models

1. User clicks on the model menu button.
2. Application offers a list of math models in drop down menu.
3. User selects the required mathematical model.
4. Application switches the page.
5. Application waits for the input of the values suitable to the selected model.

3b. Using the new model

1. User fills in the required values into the textboxes and presses “calc”.

2. Application calculates a prediction using the selected mathematical model.

3. Application presents the predicted results of future populations.

2 - Project Predictions in Graph.

**Primary Actor:** User

**Stakeholders and Interests:**

- User: Wants an accurate and easy to use application that will show them the effects of certain solutions to the unbalanced nature preserve problem.

- Company: an application which implements the necessary mathematical models which calculate the effects of certain conditions on the current situation and thus showing the best possible solution to the described problem.

**Precondition:** Prediction has been made.

**Postcondition:** There will be a graph representing the populations per year.

**Main Success Scenario (or Basic Flow):**

1. Application draws the lines for each animal in different colors.

2. Application projects the lines onto a XY line graph. (x = years, y = population)

3. User clicks on a certain point (year) on one of the lines.

4. Application presents the exact populations of each animal for the point (year) selected in step 3.

**Extensions (or Alternative Flows):**

\*a. At any time, System fails:

Assuming System crashes after clicking a line on the graph.

1. User closes the application and opens it again.

2. Application starts up.

3. User attempts the calculation again.

2a. Invalid value:

1. Application sends a pop-up message stating “Please insert a number greater than 0”.

3 - Alter initial data. (capacity and starting amount of each animal)

**Primary Actor:** User

**Stakeholders and Interests:**

- User: Wants an accurate and easy to use application that will show them the effects of certain solutions to the unbalanced nature preserve problem.

- Company: an application which implements the necessary mathematical models which calculate the effects of certain conditions on the current situation and thus showing the best possible solution to the described problem.

**Precondition:**

**Postcondition:**

**Main Success Scenario (or Basic Flow):**

1. User clicks the “change initial data” button.

2. Application opens separate window with input boxes and an “ok” button.

3. Application waits for user input.

4. User fills in the new initial data required and presses “ok”.

5. Application changes its initial data and closes the window created in step 2.

**Extensions (or Alternative Flows):**

\*a. At any time, System fails:

Assuming System crashes after attempting to change initial data

1. User closes the application and opens it again.

2. Application starts up.

3. User attempts to change the initial data once again.

2a. Invalid input:

1. Application presents a message in red-bold-font at the top of the window stating “One or more fields are invalid. Please insert a number greater than 0”.
2. User finds the insufficient input and inserts the proper changes.
3. Application changes its initial data and closes the change initial data window.

2b. Null Input:

1. A- Application makes no changes to its initial data and closes change initial data window.

B- 1. User inserts 0 into a text field as input.

2. Application sends a pop-up message stating “Are you sure you meant 0, this is different to entering nothing”.

4 - Store and load the results of a prediction to and from a file.

**Primary Actor:** User

**Stakeholders and Interests:**

- User: Wants an accurate and easy to use application that will show them the effects of certain solutions to the unbalanced nature preserve problem.

- Company: an application which implements the necessary mathematical models which calculate the effects of certain conditions on the current situation and thus showing the best possible solution to the described problem.

**Precondition:** Prediction has been made.

**Postcondition:** A record of your saved predictions will be found on your computer as a text file.

**Main Success Scenario (or Basic Flow):**

1. User clicks the “change initial data” button.

2. Application opens pop-up window with an input boxes and an “ok” button.

3. Application asks for the path you would like to store the prediction.

3. Application waits for user input.

4. User fills in his preferred path and presses “ok”.

5. Application creates the file and adds the prediction to the file.

**Extensions (or Alternative Flows):**

\*a. At any time, System fails:

Assuming System crashes after attempting to save a prediction

1. User closes the application and opens it again.

2. Application starts up.

3. User attempts to store the prediction once again.

2a. Invalid path:

1. Application presents a message in red-bold-font at the top of the pop-up window stating “Path cannot be found, try again”.
2. User changes path to the desktop.
3. Application creates the file and stores the prediction to the desktop.