**Project Specification Document**

VisKer

Akshita Mehta

Arjun Mohnot

G. Monisha

Siddhanth Iyer

Ayushi Agarwal

**Table of Contents**

Section Page

[1. Project Vision and Objectives 1](#_Toc2669142)

[1.1 Project Scope and Vision 1](#_Toc2669143)

[1.2 Project Goals and Objectives 2](#_Toc2669144)

[2 Project Planning 3](#_Toc2669145)

[2.1 Project Lifecycle 3](#_Toc2669146)

[2.2 Project Setup 3](#_Toc2669147)

[2.3 Stakeholders 3](#_Toc2669148)

[2.4 Project Resources 4](#_Toc2669149)

[3 Requirements (User Stories) 5](#_Toc2669150)

[3.1 Overall Description 5](#_Toc2669151)

[3.2 Users and Roles 5](#_Toc2669152)

[3.3 Use Case Diagrams 6](#_Toc2669153)

[3.4 User Stories (Requirements) 6](#_Toc2669154)

[3.5 Constraints and Limitations 13](#_Toc2669155)

[Definitions and Acronyms 14](#_Toc2669156)

# Project Vision and Objectives

## Project Scope and Vision

One of the most debated topics in deep learning is how to interpret and understand a trained model – particularly in the context of high-risk industries like healthcare. The term “black box” has often been associated with deep learning algorithms. How can we trust the results of a model if we can’t explain how it works? It’s a legitimate question.

Take the example of a deep learning model trained for detecting cancerous tumors. The model tells you that it is 99% sure that it has detected cancer – but it does not tell you why or how it made that decision.

Did it find an important clue in the MRI scan? Or was it just a smudge on the scan that was incorrectly detected as a tumor? This is a matter of life and death for the patient and doctors cannot afford to be wrong.

As we have seen in the cancerous tumor example, it is crucial that we know what our model is doing – and how it’s making decisions on its predictions. Typically, the reasons listed below are the most important points for a deep learning practitioner to remember:

1 Understanding how the model works

1. Assistance in Hyperparameter tuning
2. Finding out the failures of the model and getting an intuition of why they fail
3. Explaining the decisions to a consumer / end-user or a business executive

Neural nets are black boxes. In the recent years, several approaches for understanding and visualizing Convolutional Networks have been developed in the literature. They give us a way to peer into the black boxes, diagnose mis-classifications, and assess whether the network is over/under fitting.

Guided backpropagation can also be used to create trippy art, neural/texture style transfer among the list of other growing applications. The purpose of the project is to design a Visualization Tool for Keras to visualize and debug what CNN’s are learning and to show heatmaps for a large variety of models. We also intend to develop an algorithm of our own. We plan to deliver the basic version of the toolkit in our sprint 1 evaluation and further we are planning to add all combinations possible for activation function and different models, etc. The final delivery of the product will be in April 2019.

## 1.2. Project Goals and Objectives

|  |  |
| --- | --- |
| **#** | **Goal or Objective** |
| 1 | Make the system extensible – future updates like adding more models, etc. can be done easily |
| 2 | Make the system easy to support – provide good documentation, configuration/build files, administrator’s manual |
| 3 | Make the system very easy to use – users would agree that minimal to no training is needed |
| 4 | Build a prototype that demonstrates and early feedback from the customer/users |
| 5 | Have fun working on the project |
| 6 | Develop a toolkit for keras for visualizing and debugging trained keras neural net models. |
| 7 | Compatible with major backends like Theano, TensorFlow, etc. |

# 2. Project Planning

## 2.1. Project Lifecycle

We are incorporating Rapid Application Development, since it focuses more on

1.) Defining the Requirements

At the very beginning, rapid application development sets itself apart from traditional software development models. It doesn’t require you to sit with end users and get a detailed list of specifications; instead, it asks for a broad requirement. The broad nature of the requirements helps you give specific requirements at different points of the development cycle.

2.) Prototyping

This is where the actual development takes place.

Instead of following a strict set of requirements, we are creating prototypes with different features and functions as fast as they can. These prototypes are then shown to our mentor who decide what they like and what they don’t.

3.) Receiving Feedback

In this stage, feedback on what’s good, what’s not, what works, and what doesn’t is shared. Feedback isn’t limited to just pure functionality, but also visuals and interfaces.

With this feedback in mind, our prototyping continues. These two steps are repeated until we will develop a final product that can be realized which fits both the developers’ and client’s requirements.

4.) Finalizing the Software

Here, features, functions, aesthetics, and interface of the software will be finalized with the mentor. Stability, usability, and maintainability are of paramount importance before delivering to the client.

## 2.2. Project Setup

|  |  |
| --- | --- |
| **#** | **Decision Description** |
| 1 | Windows, Python, Supercomputer, Keras, Models like MobileNets, [InceptionV3](https://keras.io/applications/#inceptionv3), DenseNet, NASNets and ResNet will be used, Heatmaps from Plotly, CNN, Tensorflow, etc. |
| 2 | Standards that must be followed (default coding standard, etc.) |
| 3 | Special access privileges needed, nondisclosure forms, release to open source, etc. |
| 4 | A virtual server image will be set up at Bennett that matches the customer environment (image provided by customer) |

## 2.3. Stakeholders

|  |  |
| --- | --- |
| **Stakeholder** | **Role** |
| Dr. Deepak Garg | Sponsor |
| Dr. Anurag Goswami | Instructor |
| Dr. Madhushi Verma | Instructor and Mentor |
| Dr. Kuldeep | Instructor |
| Dr. Tanveer Ahmed | Instructor |
| Akshita Mehta | Team member |
| Arjun Mohnot | Team member |
| G. Monisha | Team member |
| Siddhanth Iyer | Team member |
| Ayushi Agarwal | Team member |

## 2.4. Project Resources

|  |  |  |
| --- | --- | --- |
| **Resource** | **Resource Description** | **Quantity** |
| Database Server | A database server provided by the sponsoring company. | 1 |
| Visker Team | Our team of students who will be the primary developers of the project. | 5 |
| Dr. Madhushi Verma | The mentor who will be able to provide us with technical assistance. | 1 |
| Windows Workstation | A Windows workstation | 1 |
| Laptop | To be used to test the software. | 2 |
| Supercomputer | For training purpose | 1 |

# 3. Requirements (User Stories)

## 3.1. Overall Description

This description may include more technical details to describe the purpose of the project.

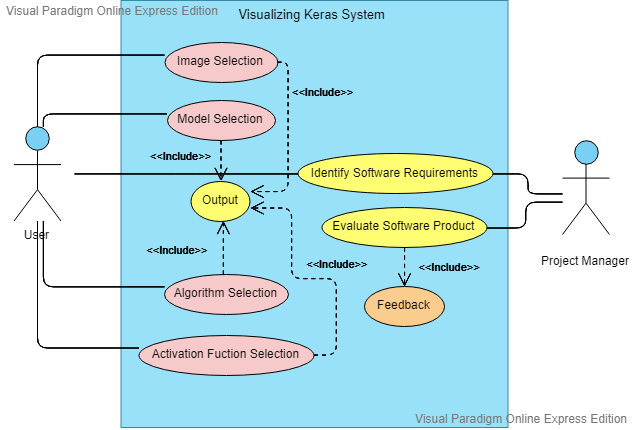
This project is an attempt to apply machine learning techniques to build an open source tool to show heat maps on a large variety of models to provide users with higher resolution. We have built a toolkit for the new networks like MobileNet or NASNet. We are trying to generate heat maps of the image by using different activation function along with which we trying to show the processing happening in different layers.

Our main motive of the project was to design a Visualization Tool for keras which allows us to view and debug what CNN’s are learning and is able to see heat maps for models like NASNet, MobileNet, etc.

## 3.2. Users and Roles

|  |  |
| --- | --- |
| **User** | **Description** |
| Developer | A visker team member or mentor who is tasked with managing the test data, creating initial machine learning models, and ultimately generating a firm process for applying these techniques to future user data. This is used for sub-stories and task needed to fulfill the true end user use cases. |

## 3.3. Use Case Diagrams



## 3.4. User Stories (Requirements)

The team is has completed the following user stories.

*As a user I want to use GUI tool kit so that I can get to know what’s happening in the hidden layers.*

*As a user I want to use different number of predictions so I can get to compare different outputs.*

*As a user I want to apply this algorithm on different models (NASNet, MobileNet, ResNet, VGG16 etc.) so I can use them according to my need.*

|  |  |  |
| --- | --- | --- |
| **ID** | **Feature name** | **Story points** |
| 1 | GUI | 3 |
| 2 | Models implementation in GUI | 3 |
| 3 | Heat maps of hidden layers | 3 |
| 4 | Developing an algorithm for all networks | 8 |
| 5 | Heat map of higher resolution | 4 |

**Estimated User Story Points: 15**

**Actual Completed User Story Points:** N/A

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **ID** | **Added** | **Description** | **Status** | **Story Points** | **Actual Equivalent Story Points** | | **% Completed** |
| 100 | Onset | As a user,  I want to be able to use a basic GUI tool kit So that I can know what’s happening in the hidden layers while training. | **C** | **6** |  | | **100%** |
| 200 | Onset | *As a user I want to use different number of predictions so I can get to compare different outputs.* | **C** | **3** |  | | **100%** |
| 300 | Onset | *As a user I want to apply this algorithm on different models (NASNet, MobileNet, ResNet, VGG16 etc.) so I can use them according to my need.* | **C** | **6** |  | | **100%** |
| **Acceptance Criteria** | | | **Verification** | | | | |
| **110** | Images should be clear and of high quality. | | **Create test cases to where the program shows the high probability to find the right object in the best quality of image.** | | | | |
| **111** | All the parameters must be selected by the user to process the image. | | **Create test case to verify that all the options must be valid before the processing starts.** | | | | |
| **112** | User should wait for some time to get the result from the application. | | **Create loader to tell the progress to the user.** | | | | |
| **ID** | **Tasks** | | | | | **Resource** | |
| 1 | Implementation of algorithm and development | | | | | **Akshita Mehta** | |
| 2 | GUI and Implementation of algorithm and development | | | | | **Arjun Mohnot** | |
| 3 | Collection of data required for different models | | | | | **G. Monisha** | |
| 4 | Efficient Algorithms development | | | | | **Ayushi Agarwal** | |
| 5 | Collection of data required for different models | | | | | **Siddhanth Iyer** | |

## 3.5. Constraints and Limitations

|  |  |
| --- | --- |
| **Constraint** | **ID** |
| Images should be clear and of good quality | 110 |
| Image should preferably be of square dimension |  |

# Definitions and Acronyms

|  |  |
| --- | --- |
| **Term** | **Definition** |
| NASNet | Neural Architecture Search Network, a family of models. that were designed automatically by learning the model architectures. |
| MobileNet | general architecture and can be used for multiple use cases. It uses depth wise separable convolutions which basically means it performs a single convolution on each color channel |
| GUI | graphical user interface is a form of user interface that allows users to interact with electronic devices |
| Keras | an open-source neural-network library written in Python. It can run on top of TensorFlow, Microsoft Cognitive Toolkit, Theano, or PlaidML. |
| Heatmap's | a representation of data in the form of a map or diagram in which data values are represented as colors |