

# Obstacles

# Project Purpose

# Stakeholder's Explanation of Project Purpose

“To design a user-friendly interface for the new deep learning toolbox TensorFlow™.”

Create a software to aid individuals in educating and creating machine learning algorithms.

Must support a drag and drop interface.

Must support TensorFlow™.

# Developer's Explanation of Project Purpose

Learn how to document and develop an application software.

Develop a software that translates visual objects manipulated by user's into machine readable and runnable code.

# Interface Design Decisions



# Map:


- Introduction
- Basics
- Demo
- Abstract
- Layers





CR-227 Product Pitch


Home Insert Design Transitions Animations Slide Show Review View


Spelling Thesaurus New Comment Delete Previous Next Show Comments Restrict Permissions

1 

2 

3 

4 

5 

**CR-227**  
side-by-side

**Battery life**

CR-227  
900 images

Fabrikam M180  
750 images

Roseware X123  
800 images

**Critic score**

**Image quality:**  
Average rating of  
9.38/10  
among top critics  
★★★★★

**Design:**  
Average rating of  
9.11/10  
★★★★★

**Features:**  
Average rating of  
8.93/10  
★★★★★


**Comments**

New

Kate Aug 6, 2014  
Also, be sure to check in with research to get the most recent critical ratings for next quarter.

Guy Gilbert 2 minutes ago  
Critical ratings have been updated. CR-227 is still the highest rated camera in its class.

Reply

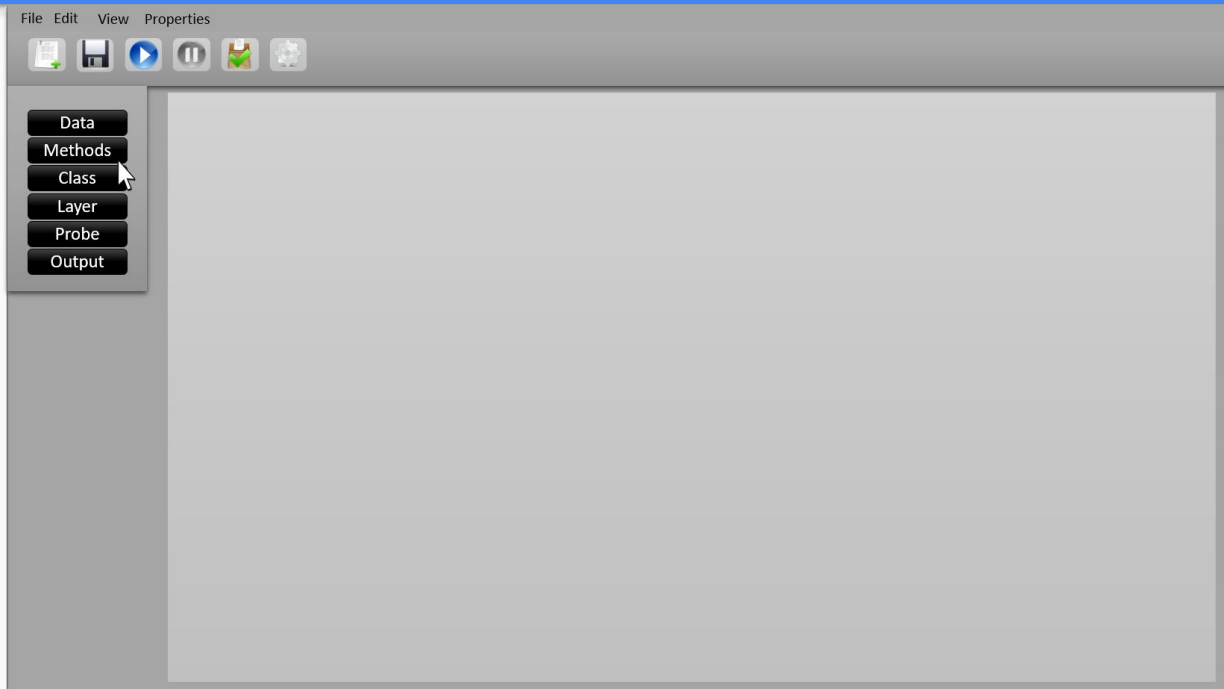


Slide 5 of 6 English (United States)

Slides Comments 14%



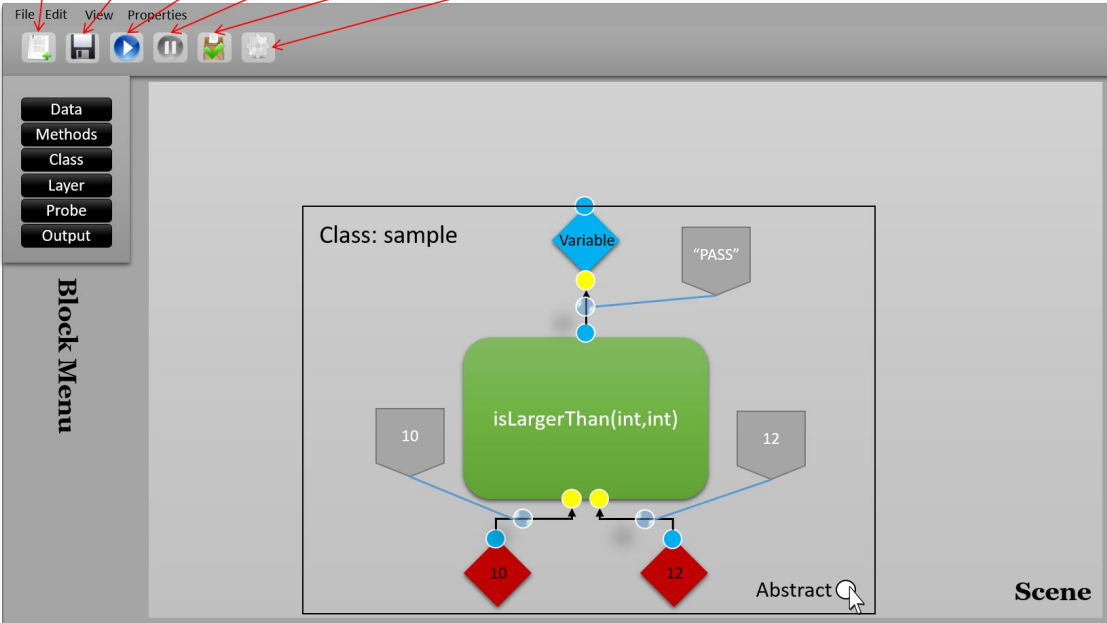
# Interface Layout (Basic)



- WYSIWYG
  - Simple
  - Easy to learn
  - Easy to use
  - Accessible

# Interface Layout (Drawn program)

New Save Run Stop Extract Settings



Class

Probe

Method

Data

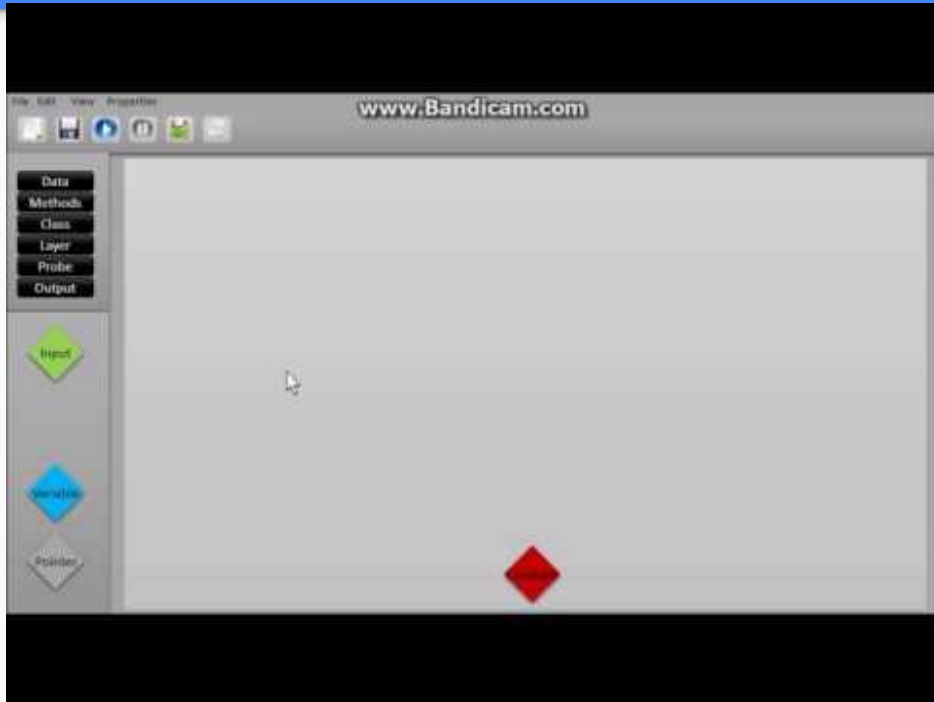
● Input

● Output

→ Channel

- New
- Save
- Run
- Stop
- Extract
- Settings
- Block Menu
- Scene
- Items

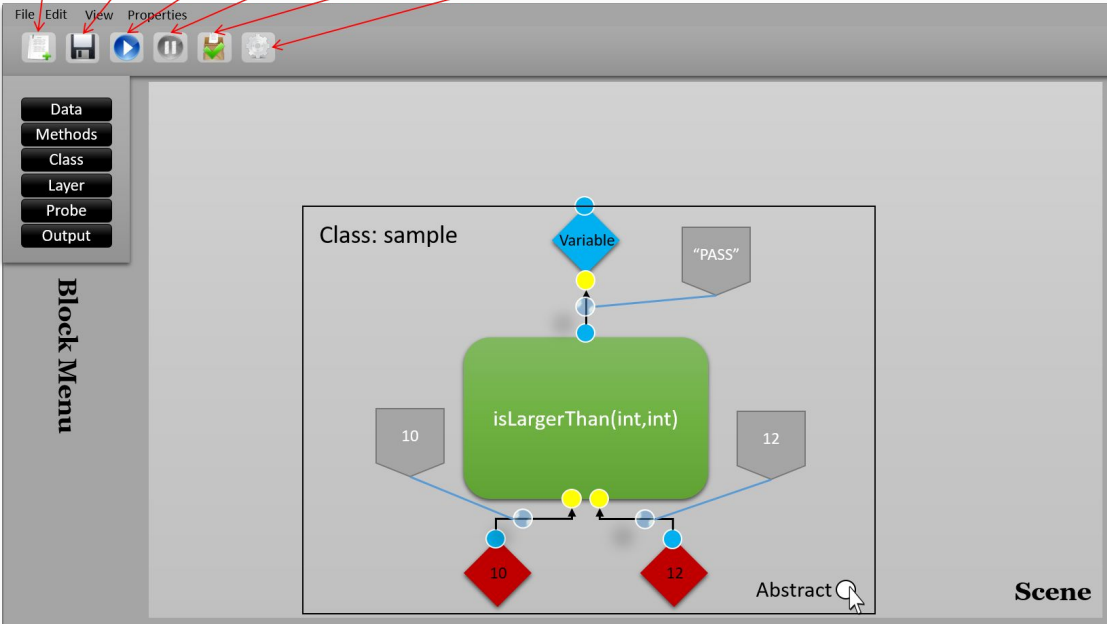
# Interface Layout (Video)



Mock-up

# Interface Layout (Inputs and putputs)

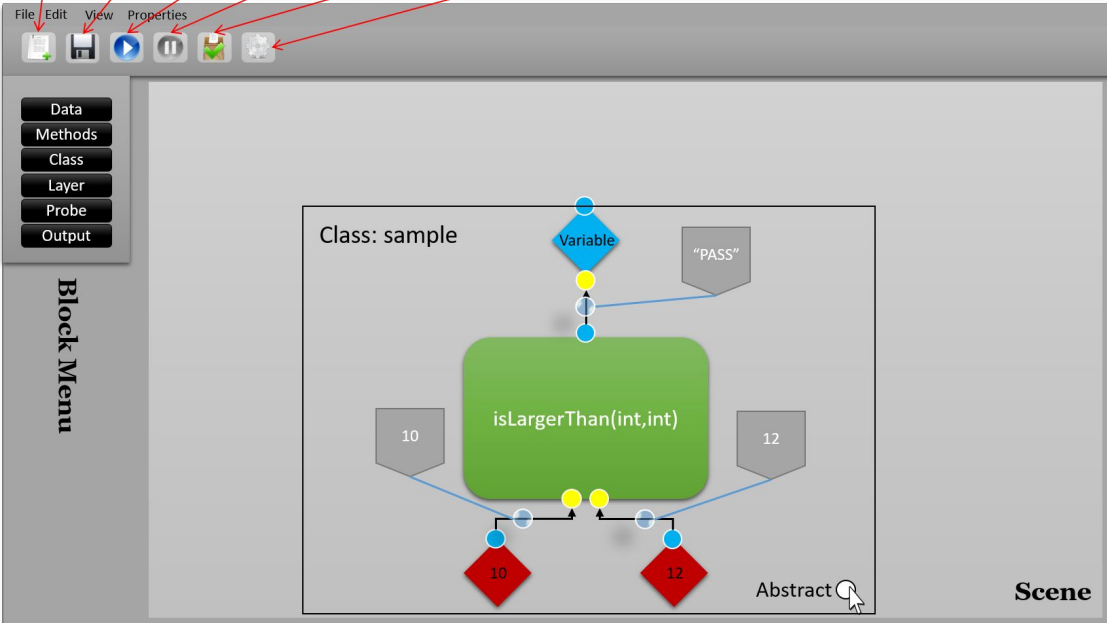
New Save Run Stop Extract Settings



- Inputs
- Outputs
- Flow

# Interface Layout (Channels)

New Save Run Stop Extract Settings



- Symbol
- Purpose
- Properties
  - Direction

Class

Probe

Method

Data

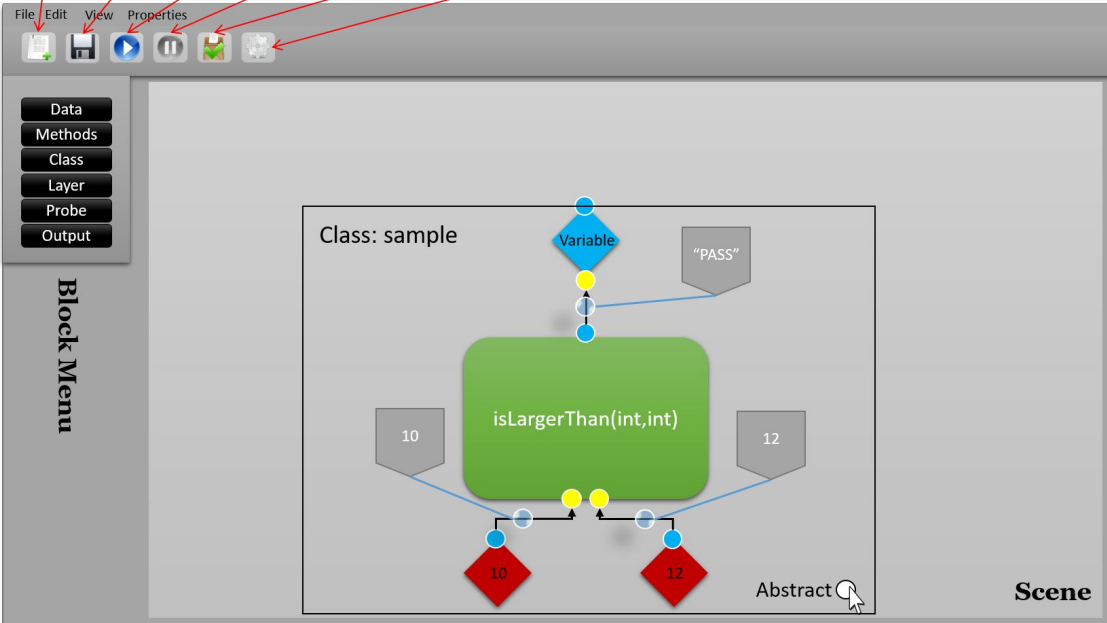
● Input

● Output

→ Channel

# Interface Layout (Probes)

New Save Run Stop Extract Settings



- Symbol
- Purpose
- Properties
  - Direction
  - Modifiable

Class

Probe

Method

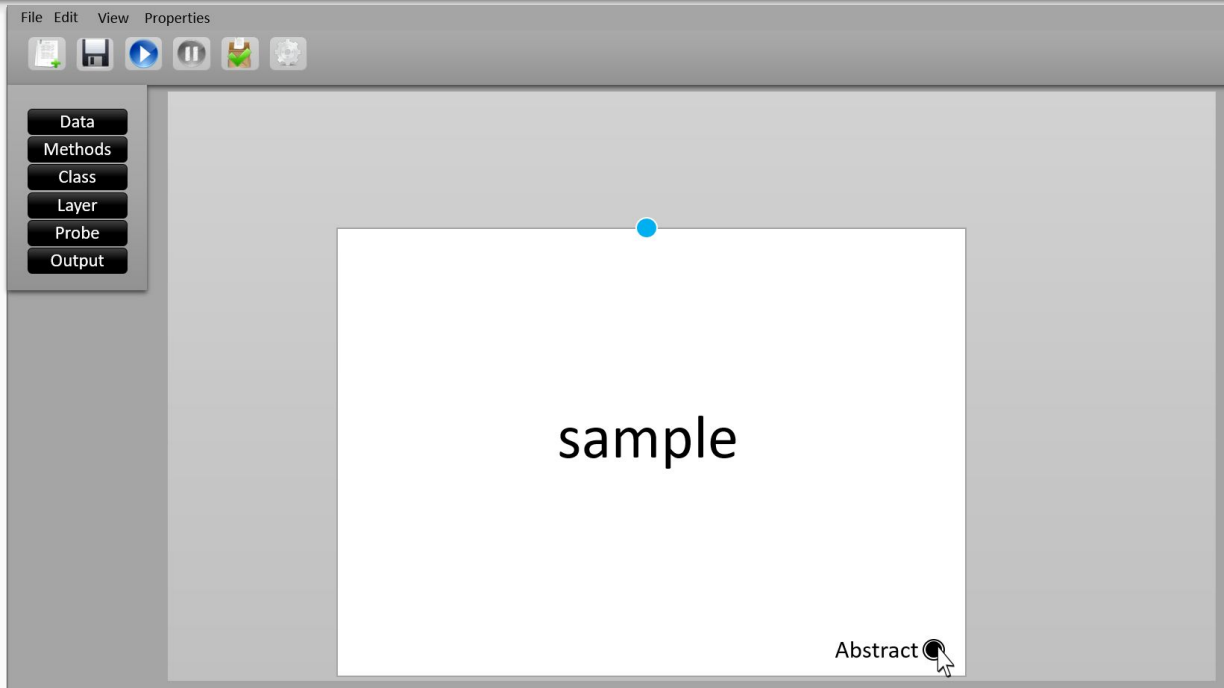
Data

● Input

● Output

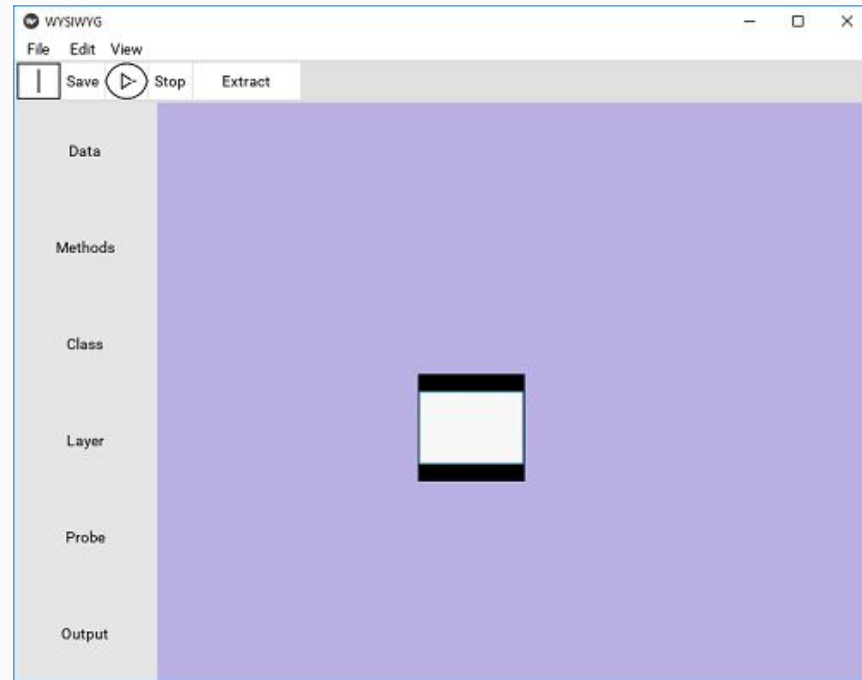
→ Channel

# Interface Layout (After Abstract is clicked)



- Symbol
- Purpose
- Properties
  - Activate and deactivate

# Current GUI





# Obstacles

# Meeting Times

## Problem:

- Scheduling meetings with Dr. Li
- Very busy and unpredictable schedule
- Need sign off signatures / review for multiple papers

## Solution:

- Constant email contact
- Proposing various meeting times
- Drop by whenever team was near Dr. Li's office
- Designated group member to handle contact

# Communication

## Problem:

- Difficulty maintaining constant and reliable contact between team members
- Different schedules and other classes
- Contribution to papers

## Solution:

- Github
- Google Drive
- Email
- Phone
- Slack
- Overleaf
- Designated meeting area

# Assignment Formatting

## Problem:

- Including images in LaTeX
- Requirements Document
- Tech Review
- Gantt Chart

## Solution:

- Meeting with TA
- Meeting with Dr. Li
- Packages included on Github
- Documentation on Gantt Chart implementation

# Understanding Assigned Tasks

## Problem:

- Varying ideas about the implementation of the project
- Arbitrary design requirements

## Solution:

- Multiple meetings with Dr. Li to discuss overall project design
- Mock up design demonstrating understanding

# Obstacles with Respect to GUI Toolkit

# Documentation Issues

- Sparse examples
- Very basic manual pages
- Spending a lot of time looking at open source software and online tutorials

# Drag and Drop Obstacles

- Kivy toolkit has a drag and drop support feature
- Does not have clear means to set bounding boxes with collisions
- Trying to figure out how to develop algorithm to create Kivy “Scatter” widgets dynamically



# File Formatting Issue

- Kivy works similar to cascading style-sheets in HTML
- Uses separate files for organization of objects on a window
- Unclear as to how the files interact and how dependencies work due to lacking documentation

# Obstacles with Respect to Core Code

# Translator Issues

- Conversion from source language to destination language
- Not reinvent the wheel