

Ryan Krawczyk, Gwyneth Steele, Hannah Ma

Final Project Single File Submission

Presentation:

Cancer Treatment Toolbox

Treatments Trials

Cancer

By: Ryan Krawczyk, Gwyneth Steele, and Hannah Ma

1

Biomedical Need

Improve the delivery of cancer research to the public

"More than half of cancer patients reported limited knowledge about the newest treatment options"

Difficultly recruiting patients for clinical trials due to lack of knowledge and public access to candidates

- Poor patient enrollment delays or even disqualifies clinical trials³
- Lack of diversity in clinical trials⁴


"70% of patients would be willing to participate in clinical trials, but less than 5% of patients actually do."

2

Background

Targets for cancer therapy

1. Growth Factor Receptors
2. Signaling Kinases
3. Transcription Factors



The diagram illustrates a cell membrane with various receptors and signaling molecules. A ligand (red sphere) binds to a receptor (blue and red structure), activating a cascade of proteins. Key components include Ras (yellow sphere), Raf (orange sphere), MEK (green sphere), and ERK (red sphere). The pathway leads to the activation of a transcription factor (blue and red structure), which then enters the nucleus to regulate gene expression. Labels include: Ligand, Receptor, Ras, Raf, MEK, ERK, Transcription Factor, and Gene Expression.

3

The composite image illustrates the process and applications of drug repurposing. The top section features a timeline titled "Drug Repurposing development" with three phases: 0-4 years (Drug bank, Drug bank, Drug bank, Drug bank, Drug bank, Drug bank), 5-7 years (Drug bank, Drug bank, Drug bank, Drug bank, Drug bank, Drug bank), and 7-12 years (Drug bank, Drug bank, Drug bank, Drug bank, Drug bank, Drug bank). Below the timeline, a list of repurposing methods includes: 1. Drug bank, 2. Drug bank, 3. Drug bank, 4. Drug bank, 5. Drug bank, 6. Drug bank, 7. Drug bank, 8. Drug bank, 9. Drug bank, 10. Drug bank, 11. Drug bank, 12. Drug bank, 13. Drug bank, 14. Drug bank, 15. Drug bank, 16. Drug bank, 17. Drug bank, 18. Drug bank, 19. Drug bank, 20. Drug bank, 21. Drug bank, 22. Drug bank, 23. Drug bank, 24. Drug bank, 25. Drug bank, 26. Drug bank, 27. Drug bank, 28. Drug bank, 29. Drug bank, 30. Drug bank, 31. Drug bank, 32. Drug bank, 33. Drug bank, 34. Drug bank, 35. Drug bank, 36. Drug bank, 37. Drug bank, 38. Drug bank, 39. Drug bank, 40. Drug bank, 41. Drug bank, 42. Drug bank, 43. Drug bank, 44. Drug bank, 45. Drug bank, 46. Drug bank, 47. Drug bank, 48. Drug bank, 49. Drug bank, 50. Drug bank, 51. Drug bank, 52. Drug bank, 53. Drug bank, 54. Drug bank, 55. Drug bank, 56. Drug bank, 57. Drug bank, 58. Drug bank, 59. Drug bank, 60. Drug bank, 61. Drug bank, 62. Drug bank, 63. Drug bank, 64. Drug bank, 65. Drug bank, 66. Drug bank, 67. Drug bank, 68. Drug bank, 69. Drug bank, 70. Drug bank, 71. Drug bank, 72. Drug bank, 73. Drug bank, 74. Drug bank, 75. Drug bank, 76. Drug bank, 77. Drug bank, 78. Drug bank, 79. Drug bank, 80. Drug bank, 81. Drug bank, 82. Drug bank, 83. Drug bank, 84. Drug bank, 85. Drug bank, 86. Drug bank, 87. Drug bank, 88. Drug bank, 89. Drug bank, 90. Drug bank, 91. Drug bank, 92. Drug bank, 93. Drug bank, 94. Drug bank, 95. Drug bank, 96. Drug bank, 97. Drug bank, 98. Drug bank, 99. Drug bank, 100. Drug bank.

The middle section, titled "DRUG REPURPOSING IN CANCER THERAPEUTICS", shows a central cluster of red spheres representing cancer cells. Surrounding this cluster are various drug molecules and their mechanisms of action, including: 1. Inhibiting cell growth, 2. Inducing cell death, 3. Inhibiting cell division, 4. Inducing cell differentiation, 5. Inhibiting cell migration, 6. Inducing cell apoptosis, 7. Inhibiting cell proliferation, 8. Inducing cell senescence, 9. Inhibiting cell invasion, 10. Inducing cell necrosis, 11. Inhibiting cell metastasis, 12. Inducing cell autophagy, 13. Inhibiting cell angiogenesis, 14. Inducing cell immunogenic cell death, 15. Inhibiting cell epithelial-mesenchymal transition, 16. Inducing cell ferroptosis, 17. Inhibiting cell hypoxia, 18. Inducing cell oxidative stress, 19. Inhibiting cell DNA damage, 20. Inducing cell telomerase inhibition, 21. Inhibiting cell telomerase activity, 22. Inducing cell telomerase expression, 23. Inhibiting cell telomerase function, 24. Inducing cell telomerase stability, 25. Inhibiting cell telomerase localization, 26. Inducing cell telomerase recruitment, 27. Inhibiting cell telomerase activation, 28. Inducing cell telomerase inactivation, 29. Inhibiting cell telomerase regulation, 30. Inducing cell telomerase expression, 31. Inhibiting cell telomerase activity, 32. Inducing cell telomerase expression, 33. Inhibiting cell telomerase activity, 34. Inducing cell telomerase expression, 35. Inhibiting cell telomerase activity, 36. Inducing cell telomerase expression, 37. Inhibiting cell telomerase activity, 38. Inducing cell telomerase expression, 39. Inhibiting cell telomerase activity, 40. Inducing cell telomerase expression, 41. Inhibiting cell telomerase activity, 42. Inducing cell telomerase expression, 43. Inhibiting cell telomerase activity, 44. Inducing cell telomerase expression, 45. Inhibiting cell telomerase activity, 46. Inducing cell telomerase expression, 47. Inhibiting cell telomerase activity, 48. Inducing cell telomerase expression, 49. Inhibiting cell telomerase activity, 50. Inducing cell telomerase expression, 51. Inhibiting cell telomerase activity, 52. Inducing cell telomerase expression, 53. Inhibiting cell telomerase activity, 54. Inducing cell telomerase expression, 55. Inhibiting cell telomerase activity, 56. Inducing cell telomerase expression, 57. Inhibiting cell telomerase activity, 58. Inducing cell telomerase expression, 59. Inhibiting cell telomerase activity, 60. Inducing cell telomerase expression, 61. Inhibiting cell telomerase activity, 62. Inducing cell telomerase expression, 63. Inhibiting cell telomerase activity, 64. Inducing cell telomerase expression, 65. Inhibiting cell telomerase activity, 66. Inducing cell telomerase expression, 67. Inhibiting cell telomerase activity, 68. Inducing cell telomerase expression, 69. Inhibiting cell telomerase activity, 70. Inducing cell telomerase expression, 71. Inhibiting cell telomerase activity, 72. Inducing cell telomerase expression, 73. Inhibiting cell telomerase activity, 74. Inducing cell telomerase expression, 75. Inhibiting cell telomerase activity, 76. Inducing cell telomerase expression, 77. Inhibiting cell telomerase activity, 78. Inducing cell telomerase expression, 79. Inhibiting cell telomerase activity, 80. Inducing cell telomerase expression, 81. Inhibiting cell telomerase activity, 82. Inducing cell telomerase expression, 83. Inhibiting cell telomerase activity, 84. Inducing cell telomerase expression, 85. Inhibiting cell telomerase activity, 86. Inducing cell telomerase expression, 87. Inhibiting cell telomerase activity, 88. Inducing cell telomerase expression, 89. Inhibiting cell telomerase activity, 90. Inducing cell telomerase expression, 91. Inhibiting cell telomerase activity, 92. Inducing cell telomerase expression, 93. Inhibiting cell telomerase activity, 94. Inducing cell telomerase expression, 95. Inhibiting cell telomerase activity, 96. Inducing cell telomerase expression, 97. Inhibiting cell telomerase activity, 98. Inducing cell telomerase expression, 99. Inhibiting cell telomerase activity, 100. Inducing cell telomerase expression.

4

Goals

Create a new **user-friendly** interface that consolidates treatment options for cancer patients so they are educated on all of their options and can choose their preferred path.

- Up to date treatment options (databases are updated as new treatments are released and withdrawn)

Impact

Increase in patient survival and quality of life

Better patient recruitment in clinical trials improves relevancy of findings, resulting in more effective treatments to be available to a larger population

5

6

[illegible]

5

The screenshot shows the 'Clinical Trial Database' website. At the top, there is a navigation bar with links: Home, About Us, Search, and Contact Us. Below the navigation bar is a large heading 'Clinical Trial Database'. Underneath, a paragraph states: 'Includes only repurposing trials of licensed drug product as an anticancer agent. Collected from following sources:'. A bulleted list follows, containing: 'ClinicalTrials.gov, EU Clinical Trials Registry, World Health Organisation International Clinical Trials Registry Platform'. Below the list, a paragraph reads: 'Database from anticancerfund.org exports out as a text file'. At the bottom of the page, there is a table with multiple columns, including 'Trial ID', 'Trial Title', 'Phase', 'Status', 'Start Date', 'End Date', 'Location', 'Sponsor', 'Investigator', 'Drug', 'Dose', 'Frequency', 'Duration', 'Eligibility', 'Contact', and 'Notes'. The table contains several rows of data, with the first row being highlighted in blue.

Clinical Trial Database

Includes only repurposing trials of licensed drug product as an anticancer agent
Collected from following sources:

- ClinicalTrials.gov, EU Clinical Trials Registry, World Health Organisation International Clinical Trials Registry Platform

Database from anticancerfund.org exports out as a text file

Trial ID	Trial Title	Phase	Status	Start Date	End Date	Location	Sponsor	Investigator	Drug	Dose	Frequency	Duration	Eligibility	Contact	Notes
1	Phase I/II Study of [Drug Name] in [Cancer Type]	I/II	Recruiting	2018-01-01	2020-12-31	USA, Europe	Anticancer Fund	Dr. [Name]	[Drug Name]	[Dose]	[Frequency]	[Duration]	[Eligibility]	[Contact]	[Notes]
2	Phase II Study of [Drug Name] in [Cancer Type]	II	Completed	2017-06-01	2019-06-30	USA	Anticancer Fund	Dr. [Name]	[Drug Name]	[Dose]	[Frequency]	[Duration]	[Eligibility]	[Contact]	[Notes]
3	Phase III Study of [Drug Name] in [Cancer Type]	III	Not Started	2021-01-01	2023-12-31	USA, Europe, Asia	Anticancer Fund	Dr. [Name]	[Drug Name]	[Dose]	[Frequency]	[Duration]	[Eligibility]	[Contact]	[Notes]

Logos for the software used in the presentation: Python, Tkinter, and Pillow (PIL Fork).

Results

At endstate, we created a GUI that is fundamentally faster and more user friendly than the existing databases portal on AntiCancerFunds websites that allows the user to both scroll treatments and clinical trials in one easy to use GUI.

1

The screenshot shows the RStudio environment. The top-left pane displays the 'Treatments' data frame with columns: 'id', 'name', 'type', 'duration', 'start', 'end', 'start_date', 'end_date', 'start_time', 'end_time', 'start_datetime', 'end_datetime', 'start_iso8601', 'end_iso8601', 'start_unix', 'end_unix', 'start_iso8601_utc', 'end_iso8601_utc', 'start_unix_utc', 'end_unix_utc', 'start_iso8601_utc_offset', 'end_iso8601_utc_offset', 'start_unix_utc_offset', 'end_unix_utc_offset', 'start_iso8601_utc_offset_offset', 'end_iso8601_utc_offset_offset', 'start_unix_utc_offset_offset', 'end_unix_utc_offset_offset', 'start_iso8601_utc_offset_offset_offset', 'end_iso8601_utc_offset_offset_offset', 'start_unix_utc_offset_offset_offset', 'end_unix_utc_offset_offset_offset'. The top-right pane shows a list of 'Treatments' objects, each with a 'name' and 'type' attribute. The bottom-left pane shows the 'Treatments' data frame again, and the bottom-right pane shows the 'Treatments' list again.

Treatments		Unit	Value
Control	1	0.0000	0.0000
Control	2	0.0000	0.0000
Control	3	0.0000	0.0000
Control	4	0.0000	0.0000
Control	5	0.0000	0.0000
Control	6	0.0000	0.0000
Control	7	0.0000	0.0000
Control	8	0.0000	0.0000
Control	9	0.0000	0.0000
Control	10	0.0000	0.0000
Control	11	0.0000	0.0000
Control	12	0.0000	0.0000
Control	13	0.0000	0.0000
Control	14	0.0000	0.0000
Control	15	0.0000	0.0000
Control	16	0.0000	0.0000
Control	17	0.0000	0.0000
Control	18	0.0000	0.0000
Control	19	0.0000	0.0000
Control	20	0.0000	0.0000
Control	21	0.0000	0.0000
Control	22	0.0000	0.0000
Control	23	0.0000	0.0000
Control	24	0.0000	0.0000
Control	25	0.0000	0.0000
Control	26	0.0000	0.0000
Control	27	0.0000	0.0000
Control	28	0.0000	0.0000
Control	29	0.0000	0.0000
Control	30	0.0000	0.0000
Control	31	0.0000	0.0000
Control	32	0.0000	0.0000
Control	33	0.0000	0.0000
Control	34	0.0000	0.0000
Control	35	0.0000	0.0000
Control	36	0.0000	0.0000
Control	37	0.0000	0.0000
Control	38	0.0000	0.0000
Control	39	0.0000	0.0000
Control	40	0.0000	0.0000
Control	41	0.0000	0.0000
Control	42	0.0000	0.0000
Control	43	0.0000	0.0000
Control	44	0.0000	0.0000
Control	45	0.0000	0.0000
Control	46	0.0000	0.0000
Control	47	0.0000	0.0000
Control	48	0.0000	0.0000
Control	49	0.0000	0.0000
Control	50	0.0000	0.0000
Control	51	0.0000	0.0000
Control	52	0.0000	0.0000
Control	53	0.0000	0.0000
Control	54	0.0000	0.0000
Control	55	0.0000	0.0000
Control	56	0.0000	0.0000
Control	57	0.0000	0.0000
Control	58	0.0000	0.0000
Control	59	0.0000	0.0000
Control	60	0.0000	0.0000
Control	61	0.0000	0.0000
Control	62	0.0000	0.0000
Control	63	0.0000	0.0000
Control	64	0.0000	0.0000
Control	65	0.0000	0.0000
Control	66	0.0000	0.0000
Control	67	0.0000	0.0000
Control	68	0.0000	0.0000
Control	69	0.0000	0.0000
Control	70	0.0000	0.0000
Control	71	0.0000	0.0000
Control	72	0.0000	0.0000
Control	73	0.0000	0.0000
Control	74	0.0000	0.0000
Control	75	0.0000	0.0000
Control	76	0.0000	0.0000
Control	77	0.0000	0.0000
Control	78	0.0000	0.0000
Control	79	0.0000	0.0000
Control	80	0.0000	0.0000
Control	81	0.0000	0.0000
Control	82	0.0000	0.0000
Control	83	0.0000	0.0000
Control	84	0.0000	0.0000
Control	85	0.0000	0.0000
Control	86	0.0000	0.0000
Control	87	0.0000	0.0000
Control	88	0.0000	0.0000
Control	89	0.0000	0.0000
Control	90	0.0000	0.0000
Control	91	0.0000	0.0000
Control	92	0.0000	0.0000
Control	93	0.0000	0.0000
Control	94	0.0000	0.0000
Control	95	0.0000	0.0000
Control	96	0.0000	0.0000
Control	97	0.0000	0.0000
Control	98	0.0000	0.0000
Control	99		

13

Discussion

Limitations:

- Must be run through Visual Studio or similar platform

Future Improvements:

- Downloadable app
- Rank treatment options based on both success rate and risks (severity of side effects) could create a database of patient outcomes
 - Challenges: This data is limited and difficult to find
- "More details" option that display risks of each treatment
- App notifications alert user when new treatments that fit their criteria are released

References

1. Kato-Takahara, T. (2005) What patients think: knowledge about advances in cancer diagnosis. *Journal of Translational Medicine*, 3(20).
2. Jorgens, J., de Caeste, L., Nui, A., & Ruyss, A. (2005) The Role of Clinical Trial Participation in Cancer Research: Needs, Problems, and Best Practices. *Journal of Clinical Oncology*, 23(16), 3689-3695.
3. Pflanzmann, G.F., & Schmitt, H. (2004) *Research Ethics and Governance of Clinical Trials*. Berlin: Springer.
4. Pflanzmann, G.F., & Schmitt, H., & Grottel, R. (2005) Research ethics and governance of clinical trials: a review of the literature. *Journal of Clinical Pharmacy and Therapeutics*, 30(1), 1-10.
5. Kato-Takahara, T. (2005) What patients think: knowledge about advances in cancer diagnosis. *Journal of Translational Medicine*, 3(20).
6. Kato-Takahara, T. (2005) What patients think: knowledge about advances in cancer diagnosis. *Journal of Translational Medicine*, 3(20).
7. Kato-Takahara, T. (2005) What patients think: knowledge about advances in cancer diagnosis. *Journal of Translational Medicine*, 3(20).
8. Kato-Takahara, T. (2005) What patients think: knowledge about advances in cancer diagnosis. *Journal of Translational Medicine*, 3(20).
9. Kato-Takahara, T. (2005) What patients think: knowledge about advances in cancer diagnosis. *Journal of Translational Medicine*, 3(20).
10. Kato-Takahara, T. (2005) What patients think: knowledge about advances in cancer diagnosis. *Journal of Translational Medicine*, 3(20).

Gui.py

```

C: > Users > sport > Dropbox > bmes550.RyanKrawczyk.rk836 > Final > gui.py > GUI > _init_
1  #import sys,os; sys.path.append(os.environ['BMESAHMETDIR']); import bmes
2
3  from tkinter import *
4  from tkinter.font import Font
5  from PIL import ImageTk, Image
6  from tkinter import ttk
7  from data_read import*
8  import re
9
10 class GUI:
11     def __init__(self) -> None:
12         self.master = Tk()
13         self.master.title("Cancer Treatment TOOLBOX v1.0")
14         self.master.geometry("1100x700")
15         self.master.configure(bg="black")
16         self.baseDIR = "C:\\Users\\sport\\Dropbox\\bmes550.RyanKrawczyk.rk836\\Final\\"
17         self.createMainPage()
18         self.createTreatmentPage()
19         self.createTrialsPage()
20         self.showMainPage()
21
22     def create_treatments_button(self):
23         self.treatment_button = Button(self.master, text="Treatments", bg="white", width=10, height=2, font=("Arial"
24
25     def create_trials_button(self):
26         self.trials_button = Button(self.master, text="Trials", bg="white", width=10, height=2, font=("Arial", 32 ),
27
28     def create_cancer_image(self):
29         self.cancer_img = PhotoImage(file=self.baseDIR+"cancer.png")
30         self.cancerImage = Label(self.master, image=self.cancer_img)
31
32     def hideTreatmentPage(self):
33

```

```

28     def create_cancer_image(self):
29         self.cancer_img = PhotoImage(file=self.baseDIR+"cancer.png")
30         self.cancerImage = Label(self.master, image=self.cancer_img)
31
32     def hideTreatmentPage(self):
33
34         self.treatmentLabel.place_forget()
35         self.tableOfTreatments.place_forget()
36         self.treescrollx.place_forget()
37         self.treescrollly.place_forget()
38
39         self.filter_btn.place_forget()
40         self.filterFDA_entry.place_forget()
41         self.filter_FDA_label.place_forget()
42         self.filter_TARGETS_entry.place_forget()
43         self.filter_TARGETS_label.place_forget()
44         self.goBack_button.place_forget()
45
46     def createTreatmentLabel(self):
47         self.treatmentLabel = Label(self.master, text="Treatments", font=("Arial", 40),bg="black",fg="white")
48
49     def createTreeView(self):
50         columns = ('DrugName', 'FDA','Targets')
51         self.tableOfTreatments = ttk.Treeview(self.master,columns=columns,show='headings')
52         s = ttk.Style()
53         s.configure('Treeview', font=('Helvetica', 16),rowheight=59 , fieldbackground = "white")
54         self.treescrollly = ttk.Scrollbar(self.master, orient="vertical", command=self.tableOfTreatments.yview)
55         self.treescrollx = ttk.Scrollbar(self.master, orient="horizontal", command=self.tableOfTreatments.xview)
56         self.tableOfTreatments.configure(xscrollcommand=self.treescrollx.set, yscrollcommand=self.treescrollly.set)
57
58     def fillTable(self):
59         tv1 = self.tableOfTreatments
60         tv1.delete(*tv1.get_children())

```



```

123         voidL = []
124
125         for med in outlist:
126             if re.search(target_filter,med[2]):
127                 voidL.append(med)
128         self.fillTable_w_input(voidL)
129
130     elif fda_filter == "N":
131         for med in medicine_list:
132             if med[1] == "N":
133                 outlist.append(med)
134
135         voidL = []
136
137         for med in outlist:
138             if re.search(target_filter,med[2]):
139                 voidL.append(med)
140         self.fillTable_w_input(voidL)
141
142     elif fda_filter == "":
143         for med in medicine_list:
144             if re.search(target_filter,med[2]):
145                 outlist.append(med)
146         self.fillTable_w_input(outlist)
147
148     """if fda_filter == "":
149         for med in medicine_list:
150             if re.search(target_filter,med[2]):
151                 print(med)"""
152
153     def filterTrials(self):
154         genderFilter = self.Gender_Entry.get()
155         NTCFilter = self.NTC_num_Entry.get()
156         CancerTypeFilter = self.CancerType_Entry.get()

```

```

153     def filterTrials(self):
154         genderFilter = self.Gender_Entry.get()
155         NTCFilter = self.NTC_num_Entry.get()
156         CancerTypeFilter = self.CancerType_Entry.get()
157         AgeFilter = self.Age_Entry.get()
158         CountryFilter = self.Country_Entry.get()
159
160         trails_list = THE_LIST_OF_CANCERS
161         the_filtered_output = trails_list
162         theNTC_filter_out = []
163         theGender_filter_out = []
164         theAge_filter_out = []
165         theCountry_filter_out = []
166         theCancerType_filter_out = []
167
168
169
170         if NTCFilter == "":
171             theNTC_filter_out = trails_list
172         else:
173             for trail in trails_list:
174                 if re.search(NTCFilter, trail[0]):
175                     theNTC_filter_out.append(trail)
176
177             #print(theNTC_filter_out)
178
179         if genderFilter == "":
180             theGender_filter_out = theNTC_filter_out
181         else:
182             for trail in theNTC_filter_out:
183                 if re.search(genderFilter, trail[1]):
184                     theGender_filter_out.append(trail)
185

```

```

186     #print(theGender_filter_out)
187
188     if AgeFilter == "":
189         theAge_filter_out = theGender_filter_out
190     else:
191         for trail in theGender_filter_out:
192             if re.search(AgeFilter, trail[2]):
193                 theAge_filter_out.append(trail)
194
195     #print(theAge_filter_out)
196
197     if CountryFilter == "":
198         theCountry_filter_out = theAge_filter_out
199     else:
200         for trail in theAge_filter_out:
201             if re.search(CountryFilter, trail[3]):
202                 theCountry_filter_out.append(trail)
203
204     if CancerTypeFilter == "":
205         theCancerType_filter_out = theCountry_filter_out
206     else:
207         for trail in theCountry_filter_out:
208             if re.search(CancerTypeFilter, trail[4]):
209                 theCancerType_filter_out.append(trail)
210
211     self.fillTrialTable(theCancerType_filter_out)
212     #print(theCancerType_filter_out)
213
214     def createTrialSort_Labels(self):
215         self.filter_trails = Button(self.master, command=self.filterTrials, bg="white", text="Filter", width=10, height=1,
216         self.NTC_num_label = Label(self.master, text="NTC Number:", bg="black", fg="white", font=('Arial', 16))
217         self.NTC_num_Entry = Entry(self.master, width=20, font=('Arial', 16))
218         self.Gender_label = Label(self.master, text="Gender:", bg="black", fg="white", font=('Arial', 16))

```

```

214     def createTrialSort_Labels(self):
215         self.filter_trails = Button(self.master, command=self.filterTrials, bg="white", text="Filter", width=10, height=1,
216         self.NTC_num_label = Label(self.master, text="NTC Number:", bg="black", fg="white", font=('Arial', 16))
217         self.NTC_num_Entry = Entry(self.master, width=20, font=('Arial', 16))
218         self.Gender_label = Label(self.master, text="Gender:", bg="black", fg="white", font=('Arial', 16))
219         self.Gender_Entry = Entry(self.master, width=10, font=('Arial', 16))
220         self.Age_label = Label(self.master, text="Age:", bg="black", fg="white", font=('Arial', 16))
221         self.Age_Entry = Entry(self.master, width=10, font=('Arial', 16))
222         self.Country_label = Label(self.master, text="Country PI:", bg="black", fg="white", font=('Arial', 16))
223         self.Country_Entry = Entry(self.master, width=16, font=('Arial', 16))
224         self.CancerType_label = Label(self.master, text="Cancer Type:", bg="black", fg="white", font=('Arial', 16))
225         self.CancerType_Entry = Entry(self.master, width=20, font=('Arial', 16))
226
227     def showTrialPage(self):
228         self.hideMainPage()
229         self.treatmentLabel.place(x=400, y=20)
230         self.tableOfTrials.place(x=50, y=300)
231         self.treescrollx.pack(side="bottom", fill="x")
232         self.treescrolly.pack(side="right", fill="y")
233         self.fillTrialTable(THE_LIST_OF_CANCERS)
234         self.NTC_num_label.place(x=60, y=200)
235         self.NTC_num_Entry.place(x=200, y=200)
236         self.Gender_label.place(x=60, y=250)
237         self.Gender_Entry.place(x=200, y=250)
238         self.CancerType_label.place(x=560, y=200)
239         self.CancerType_Entry.place(x=760, y=200)
240         self.Age_label.place(x=360, y=250)
241         self.Age_Entry.place(x=460, y=250)
242         self.Country_label.place(x=660, y=250)
243         self.Country_Entry.place(x=810, y=250)
244         self.filter_trails.place(x=850, y=120)
245         self.goBack_button.place(x=0, y=0)

```

```

240     self.Age_label.place(x=360,y=250)
241     self.Age_Entry.place(x=460,y=250)
242     self.Country_label.place(x=660,y=250)
243     self.Country_Entry.place(x=810,y=250)
244     self.filter_trails.place(x=850, y=120)
245     self.goBack_button.place(x=0,y=0)
246
247     def hideTrialPage(self):
248         self.treatmentLabel.place_forget()
249         self.tableOfTrials.place_forget()
250         self.treescrollx_.place_forget()
251         self.treescrollly_.place_forget()
252
253         self.NTC_num_label.place_forget()
254         self.NTC_num_Entry.place_forget()
255         self.Gender_label.place_forget()
256         self.Gender_Entry.place_forget()
257         self.CancerType_label.place_forget()
258         self.CancerType_Entry.place_forget()
259         self.Age_label.place_forget()
260         self.Age_Entry.place_forget()
261         self.Country_label.place_forget()
262         self.Country_Entry.place_forget()
263         self.filter_trails.place_forget()
264         self.goBack_button.place_forget()
265
266     def fillTrialTable(self,input_):
267         tv1 = self.tableOfTrials
268         tv1.delete(*tv1.get_children())
269         tv1["show"] = "headings"
270         for column in tv1["columns"]:
271             if column=="Gender" or column=="Country":
272                 tv1.column(column, width = 100)

```

```

266     def fillTrialTable(self,input_):
267         tv1 = self.tableOfTrials
268         tv1.delete(*tv1.get_children())
269         tv1["show"] = "headings"
270         for column in tv1["columns"]:
271             if column=="Gender" or column=="Country":
272                 tv1.column(column, width = 100)
273             elif column=="NTCNumber":
274                 tv1.column(column, width = 150)
275             else:
276                 tv1.column(column, width = 250)
277             tv1.heading(column, text=column)
278
279         df_rows = input_
280         for row in df_rows:
281             tv1.insert("", "end", values=row)
282
283     def createTrialsLabel(self):
284         self.treatmentLabel = Label(self.master, text="Treatments", font=("Arial", 40),bg="black",fg="white")
285
286     def createTrialsPage(self):
287         self.createTrialsLabel()
288         self.createTrialsTable()
289         self.createTrialSort_Labels()
290
291     def createTrialsTable(self):
292         columns = ('NCTNumber', 'Gender', 'Age', 'Country', 'CancerType')
293         self.tableOfTrials = ttk.Treeview(self.master,columns=columns,show='headings')
294         s = ttk.Style()

```

```

291 def createTrialsTable(self):
292     columns = ('NCTNumber', 'Gender', 'Age', 'Country', 'CancerType')
293     self.tableOfTrials = ttk.Treeview(self.master, columns=columns, show='headings')
294     s = ttk.Style()
295     s.configure('Treeview', font=('Helvetica', 16), rowheight=35, fieldbackground = "white")
296     self.treescrollly_ = ttk.Scrollbar(self.master, orient="vertical", command=self.tableOfTrials.yview)
297     self.treescrollx_ = ttk.Scrollbar(self.master, orient="horizontal", command=self.tableOfTrials.xview)
298     self.tableOfTrials.configure(xscrollcommand=self.treescrollx_.set, yscrollcommand=self.treescrollly_.set)
299
300 def createTreatmentPage(self):
301     self.createTreatmentLabel()
302     self.createTreeView()
303     self.createTreatmentSortButton()
304
305 def showTreatmentPage(self):
306     self.hideMainPage()
307     self.treatmentLabel.place(x=300, y=20)
308     self.tableOfTreatments.place(x=100, y=100)
309     self.treescrollx.pack(side="bottom", fill="x")
310     self.treescrollly.pack(side="right", fill="y")
311     self.fillTable()
312     self.filter_btn.place(x=800, y=30)
313     self.filterFDA_entry.place(x=700, y=30)
314     self.filter_FDA_label.place(x=620, y=30)
315     self.filter_TARGETS_entry.place(x=700, y=55)
316     self.filter_TARGETS_label.place(x=600, y=55)
317     self.goBack_button.place(x=0, y=0)
318
319 def createMainPage(self):
320     self.create_treatments_button()
321     self.create_trials_button()
322     self.create_cancer_image()

```

Ln 14, Col 41 Spaces: 4 UTF-8 CRLF Python 3.7

```

318
319 def createMainPage(self):
320     self.create_treatments_button()
321     self.create_trials_button()
322     self.create_cancer_image()
323     self.create_goBackButton()
324
325 def showMainPage(self):
326     self.treatment_button.place(x=200, y=200)
327     self.trials_button.place(x=500, y=200)
328     self.cancerImage.place(x=350, y = 400)
329
330 def hideMainPage(self):
331     self.treatment_button.place_forget()
332     self.trials_button.place_forget()
333     self.cancerImage.place_forget()
334
335 def mainloop(self):
336     self.master.mainloop()
337
338 def create_goBackButton(self):
339     self.goBack_button = Button(self.master, text="BACK", command=self.goBack, width=10, height=3)
340
341 def goBack(self):
342     self.hideTreatmentPage()
343     self.hideTrialPage()
344     self.showMainPage()
345
346 gui = GUI()
347 gui.mainloop()

```


Data_read.py

```
C: > Users > sport > Dropbox > bmes550.RyanKrawczyk.rk836 > Final > data_read.py > ...
1  #https://acfddata.coworks.be/cancerdrugsdb.txt
2  #https://acfddata.coworks.be/ReDO_Trials_DB.txt
3
4  import requests
5
6  info = requests.get('https://acfddata.coworks.be/cancerdrugsdb.txt')
7  splitting_val = str(info.text).split("/20")
8  THE_LIST_OF_MEDICINES = []
9
10 for i in range(1,len(splitting_val)-1):
11     voidL = []
12
13     inner_split = splitting_val[i].split(" ")
14     #print(inner_split[1],inner_split[2] ,inner_split[-2])
15     voidL.append(inner_split[1].replace("\r\n",""))
16     voidL.append(inner_split[2])
17     voidL.append(inner_split[-2])
18     THE_LIST_OF_MEDICINES.append(voidL)
19
20 #print(THE_LIST_OF_MEDICINES)
21 #print(len(THE_LIST_OF_MEDICINES))
22
23
24 info = requests.get('https://acfddata.coworks.be/ReDO_Trials_DB.txt')
25 splitting_val = str(info.text).split(" ")
26
27 input_coef = 39
28 input_self_ = 895
29
30 # required 0,9,10,31,33
31 THE_LIST_OF_CANCERS = []
32 for input_self in range(2,input_self_):
33     voidL = []
34     for i in range(input_coef*input_self,input_coef*(input_self+1)):
```

```

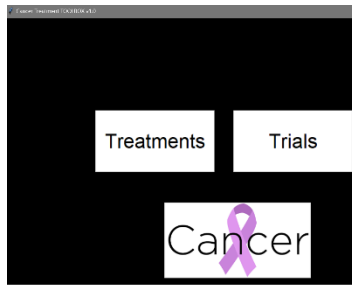
29
30 # required 0,9,10,31,33
31 THE_LIST_OF_CANCERS = []
32 for input_self in range(2,input_self_):
33     voidL = []
34     for i in range(input_coef*input_self,input_coef*(input_self+1)):
35         if i==input_coef*input_self:
36             voidL.append((splitting_val[i])[3:])
37         elif i==input_coef*input_self+8:
38             voidL.append((splitting_val[i]))
39         elif i==input_coef*input_self+9:
40             voidL.append((splitting_val[i]))
41         elif i==input_coef*input_self+32:
42             voidL.append((splitting_val[i]))
43         elif i==input_coef*input_self+34:
44             voidL.append((splitting_val[i]))
45
46
47     THE_LIST_OF_CANCERS.append(voidL)
48
49
50 #print(THE_LIST_OF_CANCERS)

```

Cancer.png



Supplemental Material and Requirements



Index.yml

```
index.yml - Notepad
File Edit View

title: GUI Cancer Toolbox
author: Ryan Krawczyk, Gwyneth Steele, Hannah Ma
abstract: This GUI utilizes the databases authored by the Anticancer Funds
Treatments and Clinical Trials Databases in order to provide a single,
lightweight, and easy to use platform for patients or researchers
to find, compare and geographically located treatments and clinical
trials with the most up to date resources available.
```