

Testing of Pinterest

Modelling and testing homework

ELTE – SPRING SEMESTER 2024

Author: Habzda Fruzsina

Tartalomjegyzék

Problem specification.....	3
Task description.....	3
Short description of the application.....	3
Analyzing the problem scope	4
Description of functions	4
Open pin	4
Save/unsave pin	5
Search for board	6
Create board.....	7
Open board.....	9
Save more pins	11
Edit board	11
Delete board.....	12
Merge board.....	13
Arrange board	13
Preconditions.....	14
FSM model.....	15
Description of vertexes	15
Variables	16
isSelectedPinSaved	16
createBoardModalWasOpenedFromProfile.....	16
boardQuantity	17
Test generating.....	18
Applied Tool.....	18
Transformation of the model	19
Applied algorithms	20
Random Walk	20
Transition Tour	21
Test unreliable reset symbols.....	22
Harmonized State Identifiers (improved version)	22
H-method	23
N-Switch Coverage	24
Results of the generations.....	25

Problem specification

Task description

This document is made for the modelling and testing subject. This version is made for the basic level homework, but I plan to also create versions for the extended level later on. For the basic-level homework, the task involves selecting a system/application/protocol and determining the areas to be modeled. Subsequently, testing should be planned, and test cases should be created based on the plan. This document serves as documentation for this process.

Short description of the application

I've chosen Pinterest¹ as my (SUT) System Under Test. This free web application is a visual search engine, mostly used by people who work in creative fields and are looking for inspiration, references, and motivation. It can also be used as a social media platform for sharing images, photos, and ideas. The shared content usually includes links to the source website.

There are two different roles in the application. Users can create 'general' or so called 'business' profiles.

Many functions help users to make collections of ideas. They can create boards and sections, save, share, and rearrange images, comment to content, communicate with others, follow topics, boards or users, and so on...

The service is currently accessible through a web browser, and apps for different OSs. Pinterest reported about 460 million monthly active users in 2023.

Although Pinterest is not open source, the company has several public repositories on Github² related to data processing, machine learning, and infrastructure.

¹ The official website of Pinterest: <https://pinterest.com/>

² A public repository of Pinterest: <https://github.com/pinterest/>

Analyzing the problem scope

As I mentioned in the chapter „Short description of the application“ there are many functionalities, use cases, roles and purposes of use. Therefore, for the completion of this task, I had to significantly reduce the tested functionalities.

I decided to focus on these functionalities:

- saving and unsaving pins
- filtering our boards before choosing one and choosing boards
- creating boards
- deleting boards
- arranging /organizing boards
- combining boards
- navigation between pages (home, profile, pin and board)

I'm going to describe these functionalities in details below.

I'm not going to test the business account, pin creating and sharing, following people or themes, chatting, section creation and management, commenting functionalities, profile, view and notification settings, or quick board creation based on AI suggestions.

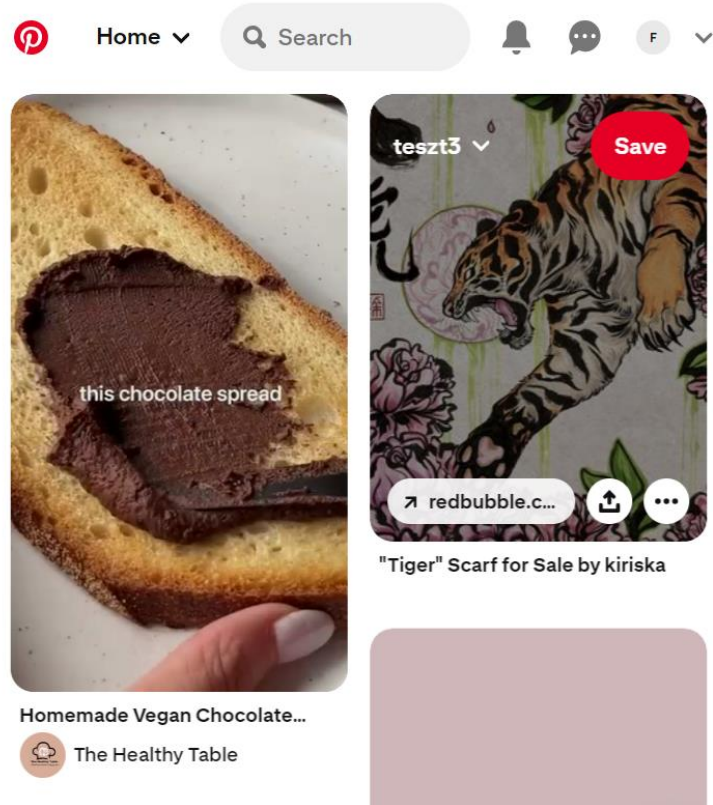
See the official user guide/documentation³ for more details on the functionalities.

Description of functions

Open pin

Pins can be opened by clicking on their image. (By hovering on the image, the 'save' button, the recommended board, and some details of the pin will appear. These are not going to be covered in my test cases.)

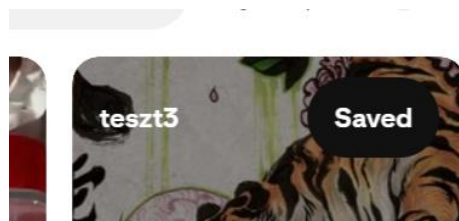
³ <https://help.pinterest.com/en/guide/all-about-pinterest>



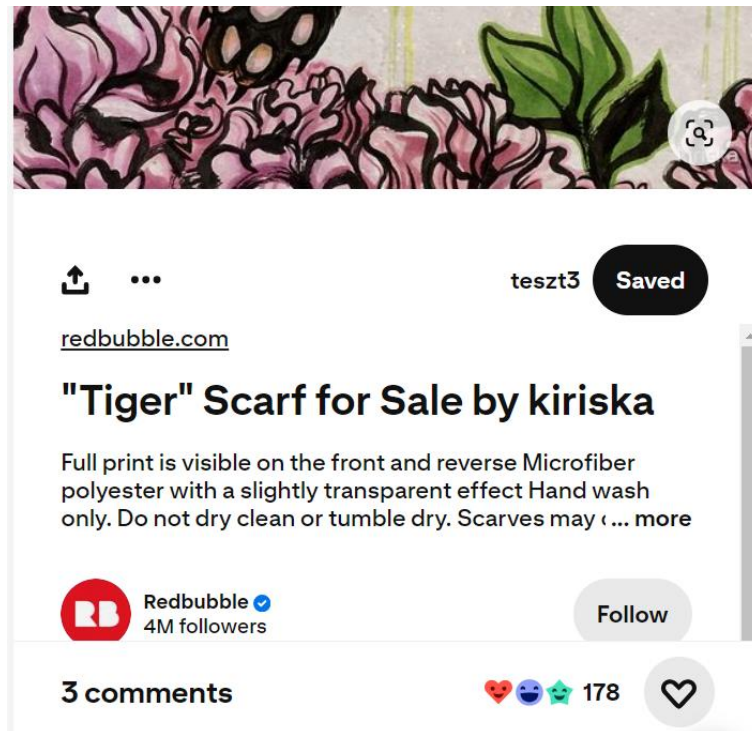
1. ábra - Home page

Save/unsave pin

Pins can be saved either opening it, and clicking on the 'save' button, or without opening it, and clicking on the 'save' button at its upperright corner. The unsaving works the same, by clicking on the same buttons. After saving a pin, the mentioned buttons change text to 'unsave', referring to the changed functionality.



2. ábra - Saved button on a saved pin

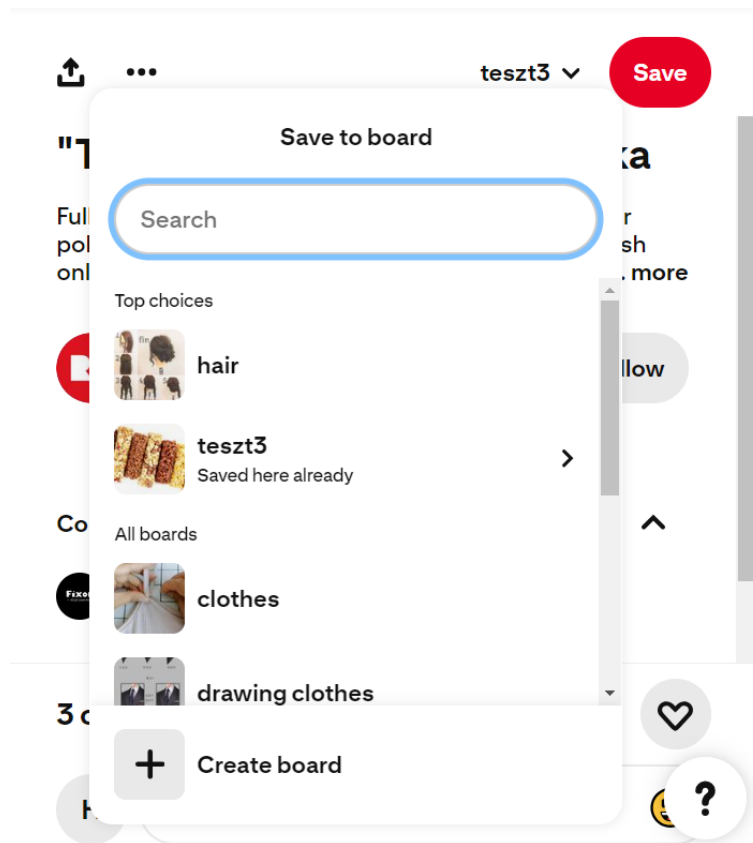


3. ábra - Opened saved pin

Search for board

We can select a board, where we'd like to save the opened pin. The application has a search modal⁴ for that, where you can type in the board's name, and it filters the boards accordingly. We can save the pin by clicking on the desired board.

⁴ by writing modal, I refer to popup window



4. ábra - Search modal

Create board

You can create new board either at the search modal, or at your profile.

At the search modal, you can click on the 'Create new board' button at the bottom.

At the profile you can click on the '+' button, and select 'Board'.



Fruzzi

habzdafruzsina

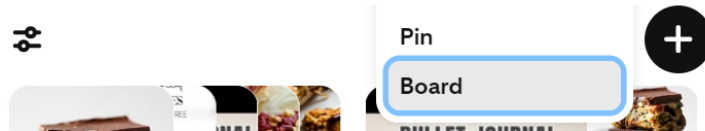
1 follower · 6 following

Share

Edit profile

Created

Saved



5. ábra - Creation options on Profile page

You have to give a non existing name to your board, to be created. If you already have a board with the given name, the system is going to give you an error message about it.

Create board

Name

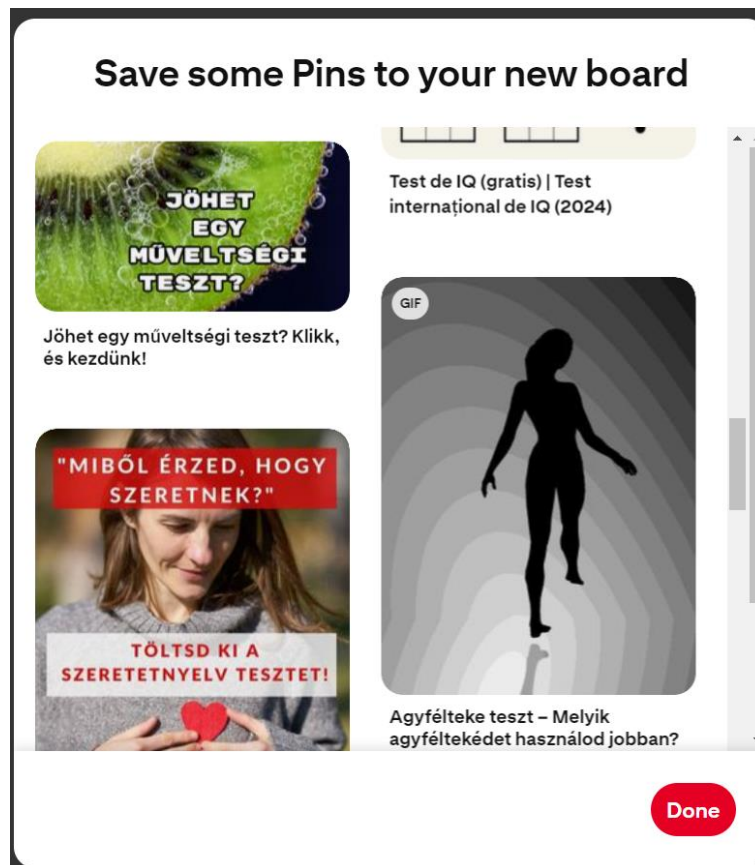
☐ **Keep this board secret**
So only you and collaborators can see it. [Learn more](#)

Create

Try a different name. You already have a board with this name.

6. ábra - Create board modal with error message

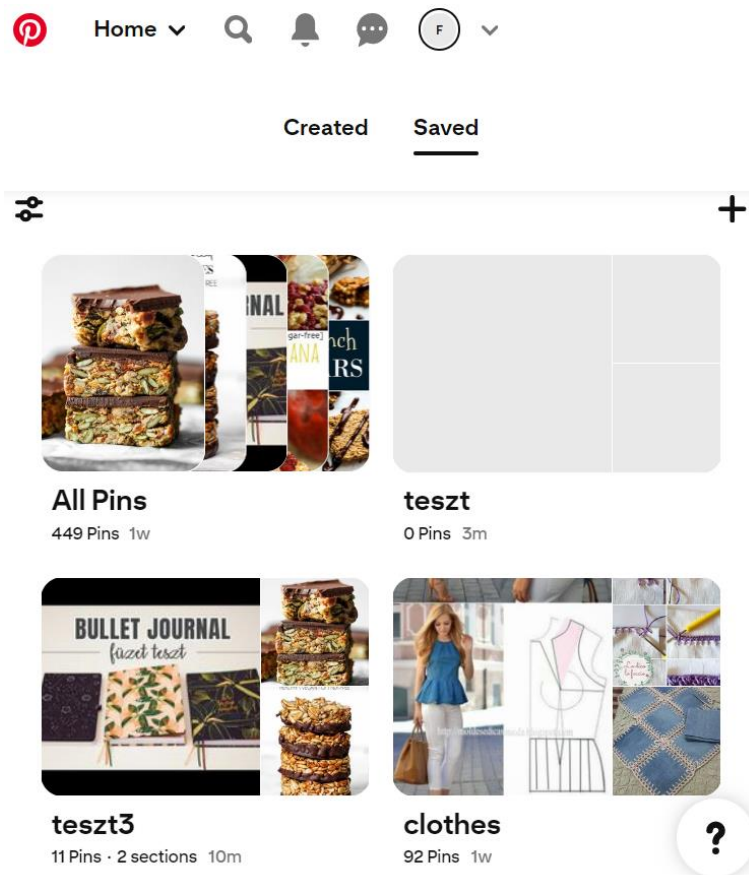
Depending on where you opened the modal, it's going to work a bit differently. For example opening it from the 'Profile', after the creation of the board, a page will appear, where you can choose new ideas for your new board. This page is not going to be opened from the search modal.



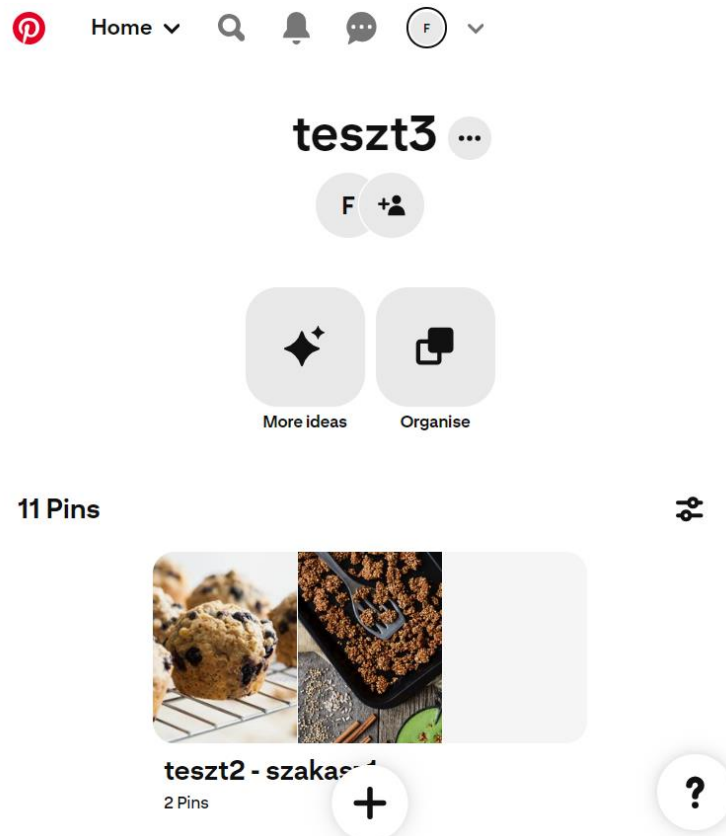
7. ábra - Save more pins/ideas to board modal

Open board

After you've navigated to your Profile, you can open your boards by clicking on them.



8. ábra - Boards on Profile page

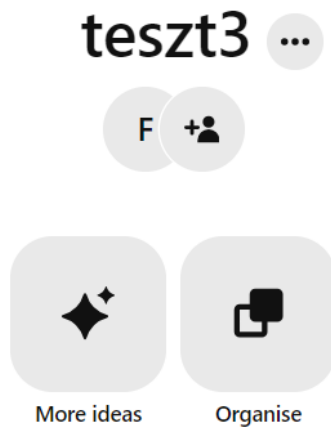


9. ábra - Page of opened "teszt3" board

After saving a pin, you can open its board by clicking on the name of the board.

[Save more pins](#)

This page can be reached from the board page, or after creating the board (as I mentioned above). Here Pinterest gives you filtered content by an AI, which is connected to a specific topic according to your interest. You can save pins here only for the selected board.

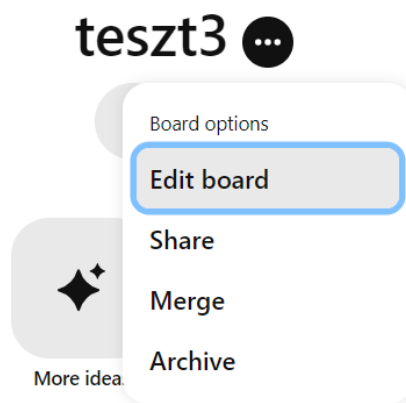


10. ábra - Options at the Page of an opened board

At the board page, you can get there by clicking on the 'More ideas' button, as showed above.

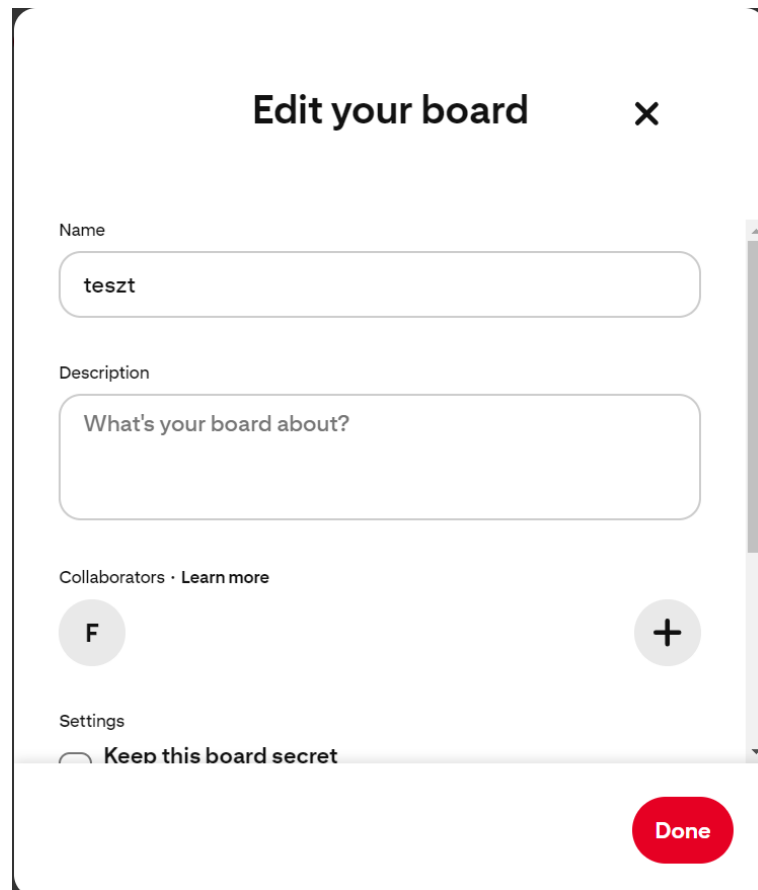
[Edit board](#)

By navigating to one of our boards, clicking on the '...' a selection opens.



11. ábra - Options selection menu at board page

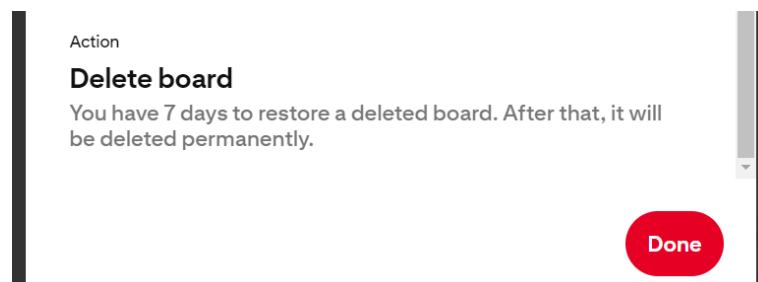
There we can choose 'edit board', so a modal opens. Here we can change its name, and edit some properties of the selected board.



12. ábra - Edit board modal

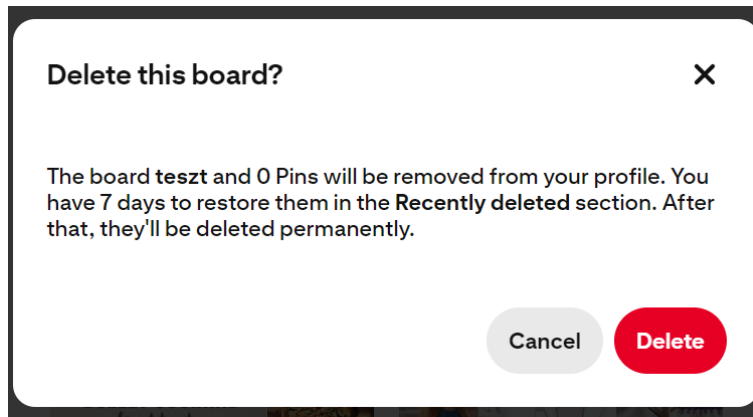
Delete board

We can delete our board by clicking on 'delete board' (action) at the bottom of 'edit board' modal.



13. ábra - Delete option at the bottom of the edit board modal

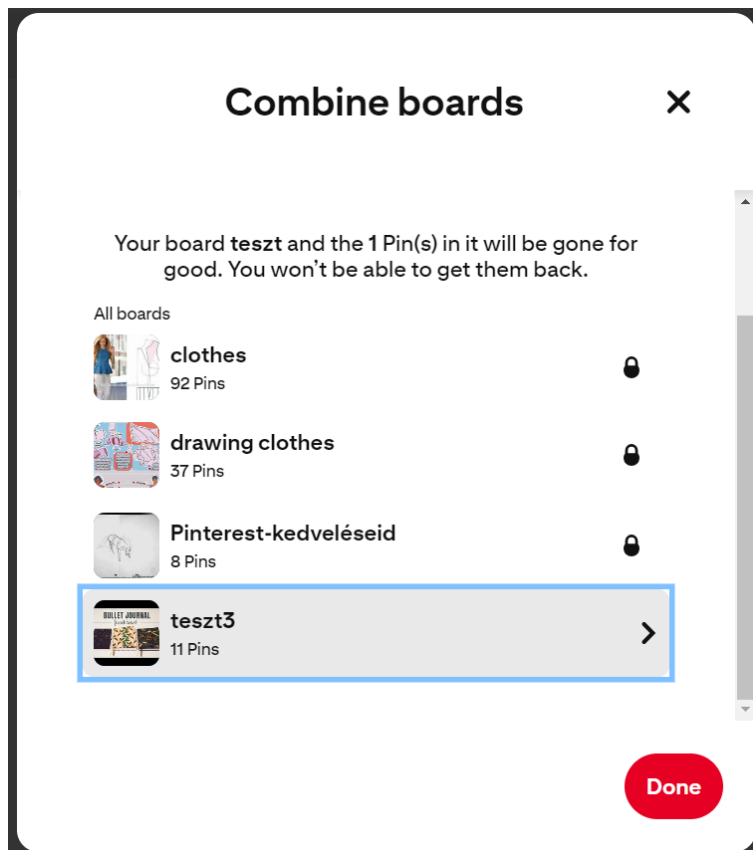
Before deleting the board de facto, the system makes sure if we really intend to delete it.



14. ábra - 'Are you sure to delete board?' modal

Merge board

'Merge boards' is under '...'. We can choose one board which we'd like to merge into. After clicking 'Done' the system is going to copy the content to the selected board, and then delete the original board. It's going to give us a message „moving to {selected boardname}\".



15. ábra - Combine boards modal

Arrange board

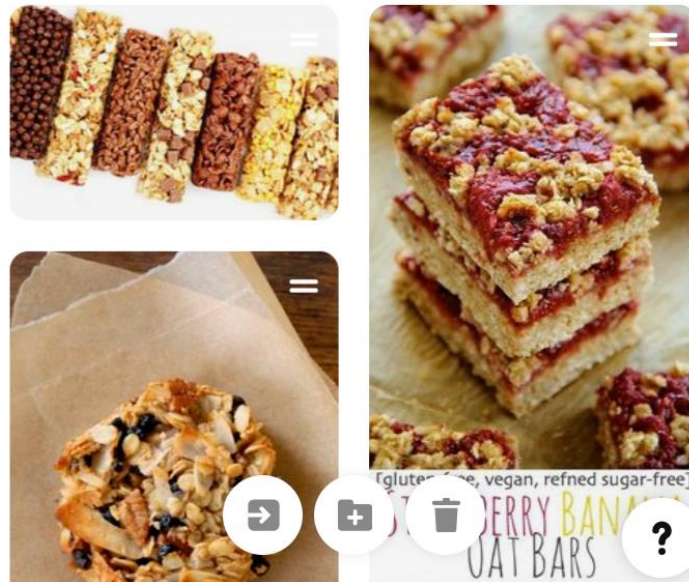
By clicking on the 'organize' button at the board page, we can start arranging our board. Drag and drop pins to change their order.



Select and reorder

Select Pins to start

Select all



16. ábra - Organize selected board page

Preconditions

To be able to test the application, we have to log into an existing 'general' profile. For a smooth start, I would recommend to use one that doesn't have any created boards or saved pins yet.⁵ If the profile already has boards, then the 'boardQuantity' variable of the model should be adapted. (By default, it starts counting from 0.)

⁵ Creating a profile is really easy and totally free. It requires an email address and takes just a few steps to complete.

FSM model

For the sake of simplicity and readability I created an EFSM (Extended Finite State Machine) instead of an FSM. I introduced variables and used them in actions and guarding conditions at some transitions.

I used GraphWalker⁶ for this purpose.

Unfortunately, it does not support input and output for the transitions, and has some functionalities, which are not included in the FSM standard. However, I tried to use it in the standard way and completed the graph manually with the missing elements. Any text editor is suitable for this task since the output of GraphWalker is a JSON file, but I personally used Visual Studio Code."

I decided to use the 'home' page as the starting point in the modal, since after logging in, we arrive at that page first.

I also added reset transitions to most of the states. While I could have added them to nearly every state, I deemed it unnecessary as it's not the primary functionality of the application. Moreover, adding resets to every state would have diminished my graph's readability.

The inputs in this model represent actions initiated by the user. Their names are self-explanatory.

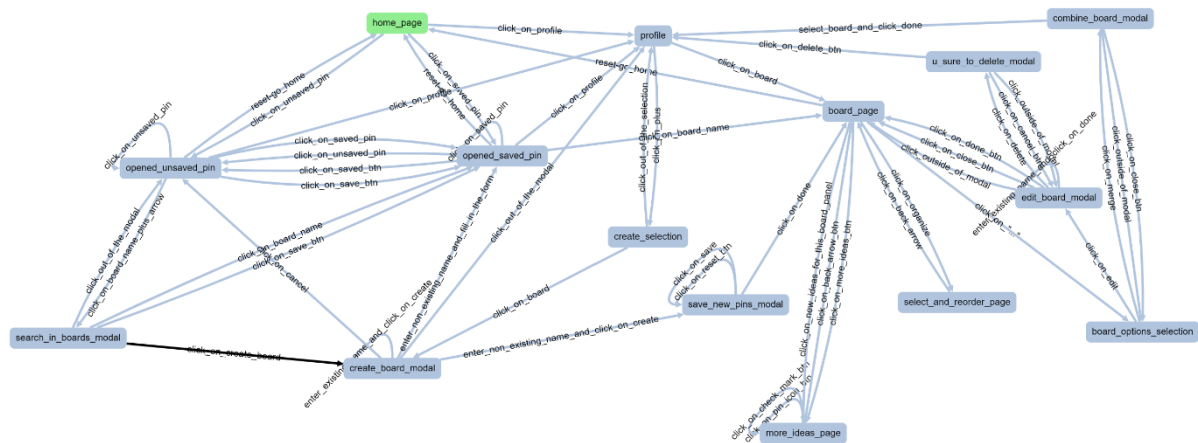
However, the outputs are not always so clear in this application. After an action, the view changes in various ways, sometimes in several different ways at once, making it difficult to track and describe. Therefore, I tried my best to give the most descriptive names to the outputs. However, when I wrote 'changedView', the application did not change the whole page, but many things changed on the view. Unfortunately, in these cases, I couldn't give them a neat name.

Description of vertexes

Vertex of the graph	Description
home_page	Home page, where several pins can be found according to our interest
opened_unsaved_pin	Opened pin page with details of the unsaved pin, and similar pins listed
search_in_boards_modal	A modal, opened from opened pin, where we can filter our existing boards to find the one, where we'd like to save the pin
opened_saved_pin	Opened pin page with details of the saved pin, and similar pins listed
profile	Profile page, with your boards listed
create_selection	Selection menu opened at profile page, with two possibilities: board, pin
save_new_pins_modal	Modal for saving new pins to a certain board, contains a list of pins
create_board_modal	A modal with a form for board creation
board_page	Page of opened board, with further options and its pins
more_ideas_page	Page for saving new ideas to a certain board, contains a list of pins
board_options_selection	Selection of options at board page

⁶ GraphWalker is an open source graphical application.
For more information: <https://graphwalker.github.io>

edit_board_modal	Edit modal, contains a form to modify the properties of the selected board (also contains delete)
u_sure_to_delete_modal	Modal for deleting board, appears after selecting delete at the edit board modal
combine_board_modal	Modal for merging two boards, contains a listing of our boards
select_and_reorder_page	Page for organizing the pins in the selected board



17. ábra - EFSM graph of the covered functionalities

Variables

I introduced a number variable and two boolean variables, and used them as guards in some cases. This was necessary because, as you can see in the 'Description of Functions' section, many options, modals, and functions can be reached from different locations in the app. Although they are almost the same, there are some differences in behavior based on the location. In my opinion, the model could be further simplified from these aspects, but it would reduce readability and cause adaptation difficulties in cases of specification changes.

isSelectedPinSaved

This variable is a simple boolean, initially set to 'false'. I introduced this to track the 'saved' property of a pin and to examine which buttons are activated and which functions are accessible. Following the action of saving or unsaving a pin, various changes occur on the viewport; such as alterations in button text and appearance, and the enabling or disabling of certain functions, etc.

createBoardModalWasOpenedFromProfile

This variable is a boolean with a default value of 'false'. It is set at two transitions: one when we'd like to create a board from our Profile page, and the other when we'd like to create a board while saving a pin.

If it is 'true', it means that we opened the modal from the Profile. In this case, a 'save pins for this board' modal will appear during the board creation process; otherwise, it won't. At the end of the process, the page of the new board will be opened.

If it's 'false', it means that we opened the modal from somewhere else (in this test model, I included only one other option: creation while pin saving). In this case, at the end of the board creation, we will end up at the original location - in our case, the 'opened pin' page.

boardQuantity

This variable is of the number type, and by default, it starts from 0. It represents the number of boards on our profile. It increases when creating a new board and decreases when deleting one or merging two. It is used as a guard condition for merging; you cannot merge boards when you don't have at least 2 of them.

Test generating

Applied Tool

For the test generation, I used the Model » Test » Relax (MTR) framework.

MTR is an open-source Model-Based Testing (MBT) framework built upon Finite State Machine (FSM) and Extended Finite State Machine (EFSM) models. It is developed at ELTE University. Test cases can be automatically generated from formal specification FSM or EFSM models to meet predefined testing objectives using a variety of test generation algorithms. This suite is then applied to the System Under Test, typically considered a black-box with unknown behavior, where the observed output sequences are compared against the expected results derived from the specification to verify conformance.

This tool can handle JSON format models created by GraphWalker. The outputs – so the test suits, converted or transformed models - are also JSON format, or dot format for Graphviz. MTR can convert EFSM to FSM, partially specified model to completely specified ones, it can add reset transitions, minimize model, and add input/output symbols.⁷

Furthermore, MTR can also help in generating adaptation code interfaces, and visualizing models and test generation results, but I'm not going to use these functions in the basic homework.

I've created two versions of EFSM. The first version⁸ includes actions that, while descriptive, cannot actually be executed. They serve to inform the reader of what the system should ideally do at those transitions. However, MTR cannot process these actions, so I've removed them in the second version⁹.

Some info of my second version EFSM model, using MTR:

```
E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -o lmi -f my_models/final_v4.json
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[U-LMI  ] [info] Model summary:
      Name: pin_saving_and_board_creating_at_pinterest
      Type: EFSM
      Number of states: 15
      Number of transitions: 56
      Density: 3.733333
      Number of input symbols: 40
      Number of output symbols: 28
      Completely specified: No
      Strongly connected: Yes
      Reset: No
      Number of variables: 3
      Number of actions: 19
      Number of guarding conditions: 7
```

18. ábra - Informatin of the EFSM model

⁷ MTR documentation: https://gitlab.inf.elte.hu/nga/ModelTestRelax/-/blob/master/docs/user_guide.md

⁸ my_models/final_v3.json

⁹ my_models/final_v4.json

Transformation of the model

Since FSM (Finite State Machine) is more ideal to most of the test generation algorithms, also considering that guard conditions are ignored when generating tests from EFSM (Extended Finite State Machine), I opted to convert my EFSM to FSM.

To carry out the mentioned conversion, I needed to introduce a new start state and transition to initialize my variables¹⁰.

```
E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -o conversion -m efsm_to_fsm -f my_models/added_start_state.json --graphviz true
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[M-FSM ] [warning] Empty string for input for transition id: cd16567f-9c7d-4f9a-bd0d-ad741c6814b5, name: start_actions
[M-FSM ] [warning] Empty string for output for transition id: cd16567f-9c7d-4f9a-bd0d-ad741c6814b5, name: start_actions
[C-EFSM ] [info] Starting transformation on model: pin_saving_and_board_creating_at_pinterest
[C-EFSM ] [warning] [create_board_modal;boardQuantity=5;createBoardModalWasOpenedFromProfile=true;isSelectedPinSaved=true]
] Dropping state from transformed graph: save_new_pins_modal;boardQuantity=6;createBoardModalWasOpenedFromProfile=true;isSelectedPinSaved=true, variable values are out of bounds
[C-EFSM ] [warning] [create_board_modal;boardQuantity=5;createBoardModalWasOpenedFromProfile=true;isSelectedPinSaved=true]
] Dropping state from transformed graph: save_new_pins_modal;boardQuantity=6;createBoardModalWasOpenedFromProfile=true;isSelectedPinSaved=true, variable values are out of bounds
[C-EFSM ] [warning] [create_board_modal;boardQuantity=5;createBoardModalWasOpenedFromProfile=true;isSelectedPinSaved=false]
e] Dropping state from transformed graph: save_new_pins_modal;boardQuantity=6;createBoardModalWasOpenedFromProfile=true;isSelectedPinSaved=false, variable values are out of bounds
[C-EFSM ] [warning] [create_board_modal;boardQuantity=5;createBoardModalWasOpenedFromProfile=true;isSelectedPinSaved=false]
e] Dropping state from transformed graph: save_new_pins_modal;boardQuantity=6;createBoardModalWasOpenedFromProfile=true;isSelectedPinSaved=false, variable values are out of bounds
[C-EFSM ] [warning] [u_sure_to_delete_modal;boardQuantity=0;createBoardModalWasOpenedFromProfile=true;isSelectedPinSaved=true]
] Dropping state from transformed graph: profile;boardQuantity=-1;createBoardModalWasOpenedFromProfile=true;isSelectedPinSaved=true, variable values are out of bounds
[C-EFSM ] [warning] [create_board_modal;boardQuantity=5;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=false]
e] Dropping state from transformed graph: opened_saved_pin;boardQuantity=6;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=true, variable values are out of bounds
[C-EFSM ] [warning] [create_board_modal;boardQuantity=5;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=false]
e] Dropping state from transformed graph: opened_saved_pin;boardQuantity=6;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=true, variable values are out of bounds
[C-EFSM ] [warning] [u_sure_to_delete_modal;boardQuantity=0;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=false]
] Dropping state from transformed graph: profile;boardQuantity=-1;createBoardModalWasOpenedFromProfile=true;isSelectedPinSaved=false, variable values are out of bounds
[C-EFSM ] [warning] [create_board_modal;boardQuantity=5;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=false]
e] Dropping state from transformed graph: opened_saved_pin;boardQuantity=6;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=true, variable values are out of bounds
[C-EFSM ] [warning] [u_sure_to_delete_modal;boardQuantity=0;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=true]
] Dropping state from transformed graph: profile;boardQuantity=-1;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=true, variable values are out of bounds
[C-EFSM ] [info] Finished computation at 2024-04-14 17:02:36.4965667, elapsed time: 0.089000(user time) seconds
[C-EFSM ] [info] All possible states visited, state count: 297, arc count: 980
[C-EFSM ] [info] finished computation at 2024-04-14 17:02:36.4965667
elapsed time: (real time)0.0887114 s
elapsed time (user time)0.089000
[U-CTRL ] [info] Transformation done, output path: converted/pin_saving_and_board_creating_at_pinterest-transformed.json
```

19. ábra - Output of the conversion to FSM

As you can observe, it excluded states that would be unreachable due to guard conditions. To prevent an explosion in the number of states in my FSM, I limited the number of boards to 5.

¹⁰ my_models/added_start_state.json

```
E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -o lmi -f converted/pin_saving_and_board_creating_at_pinterest
-transformed.json --exclude_reduced_log
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[U-LMI  ] [info] Model summary:
      Name: pin_saving_and_board_creating_at_pinterest-transformed
      Type: FSM
      Number of states: 297
      Number of transitions: 980
      Density: 3.299663
      Number of input symbols: 41
      Number of output symbols: 29
      Completely specified: No
      Strongly connected: No
      Deterministic: Yes
      Reset: No
```

20. ábra - Information of the FSM¹¹

To be able to use the generated FSM, it must be strongly connected and deterministic. To achieve this, I removed the start state and transition, and have given 'n130' as the 'startElementId'.¹²

Applied algorithms

To provide a brief overview of the applied algorithms, I included a short description from the MTR documentation for each of them, enclosed in quotes. For more detailed information, please read the MTR documentation and its references.¹³

Random Walk

(with 100% coverage)

„Starting from the initial state, in each step a transition originating from the current state is selected randomly.

Stops if one of the selected stop conditions is fulfilled:

- Given percentage of visited states is reached
- Given percentage of visited transitions is reached

Although this approach can be useful for exploratory testing, it is impractical for the functional testing of a large-scale software as the length of the test sequence can be much longer than the optimal solution.”

¹¹ converted/pin_saving_and_board_creating_at_pinterest-transformed.json

¹² converted/pinterest_fsm.json

¹³ https://gitlab.inf.elte.hu/nga/ModelTestRelax/-/blob/master/docs/user_guide.md#6-test-generation

```

E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -m Random --random_coverage_type transition --random_coverage_percent
100 -f converted/pinterest_fsm.json
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[U-LMI ] [info] Model Name: pin_saving_and_board_creating_at_pinterest-transformed, Type: FSM, Reset: No
[U-CTRL ] [info] Running Random test generation, coverage percent: 100.000000, coverage type: transition
[TG-RW ] [info] Desired transition coverage reached: 100.0%, Walk length: 469226
[TG-RW ] [info] Finished computation at 2024-04-14 17:16:43.7095831
elapsed time: (real time) 1.1858 s
elapsed time: (user time) 1.186000 s
[TG-TGR ] [info] Test generation summary written: test_summary/random_result.csv
[U-TW ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-Random
-transition-100-test_suite.json

E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -m Random --random_coverage_type transition --random_coverage_percent
100 -f converted/pinterest_fsm.json
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[U-LMI ] [info] Model Name: pin_saving_and_board_creating_at_pinterest-transformed, Type: FSM, Reset: No
[U-CTRL ] [info] Running Random test generation, coverage percent: 100.000000, coverage type: transition
[TG-RW ] [info] Desired transition coverage reached: 100.0%, Walk length: 320303
[TG-RW ] [info] Finished computation at 2024-04-14 17:17:13.8871873
elapsed time: (real time) 0.833483 s
elapsed time: (user time) 0.833000 s
[TG-TGR ] [info] Test generation summary written: test_summary/random_result.csv
[U-TW ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-Random
-transition-100-test_suite.json

E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -m Random --random_coverage_type transition --random_coverage_percent
100 -f converted/pinterest_fsm.json
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[U-LMI ] [info] Model Name: pin_saving_and_board_creating_at_pinterest-transformed, Type: FSM, Reset: No
[U-CTRL ] [info] Running Random test generation, coverage percent: 100.000000, coverage type: transition
[TG-RW ] [info] Desired transition coverage reached: 100.0%, Walk length: 209443
[TG-RW ] [info] Finished computation at 2024-04-14 17:17:18.4628829
elapsed time: (real time) 0.533258 s
elapsed time: (user time) 0.533000 s
[TG-TGR ] [info] Test generation summary written: test_summary/random_result.csv
[U-TW ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-Random
-transition-100-test_suite.json

```

21. ábra - Outputs of Random walk generations

Transition Tour

„The Transition Tour algorithm produces a test sequence that visits every transition of a reduced, deterministic, strongly connected FSM at least once and then returns to the initial state (this is the shortest tour that provides 100% state- and transition coverage). It guarantees to discover all output faults, but does not guarantee to find transfer faults.”

```

[TG-RW ] [info] Finished computation at 2024-04-14 17:17:18.4628829
           elapsed time: (real time) 0.533258 s
           elapsed time: (user time) 0.533000 s
[TG-TGR ] [info] Test generation summary written: test_summary/random_result.csv
[U-TW ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-Random-
-transition-100-test_suite.json

E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -m TT -f converted/pinterest_fsm.json
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[U-LMI ] [info] Model Name: pin_saving_and_board_creating_at_pinterest-transformed, Type: FSM, Reset: No
[U-CTRL ] [info] Running Transition Tour test generation
[TG-TT ] [info] Not Eulerian, augment
[TG-TT ] [info] Building bipartite graph
[TG-TT ] [info] Bipartite graph successfully built
[TG-TT ] [info] Creating matching
[TG-TT ] [info] Matching done
[TG-TT ] [info] Duplicating transitions according to matching
[TG-TT ] [info] Augmenting to Eulerian graph successful
[TG-TT ] [info] Ordering edges
[TG-TT ] [info] Edges ordered
[TG-TT ] [info] Generating test sequence
[TG-TT ] [info] Starting traversal
[TG-TT ] [info] Tour length: 1498
[TG-TT ] [info] Finished computation at 2024-04-14 17:20:17.8411759
           elapsed time: (real time) 1.80607 s
           elapsed time: (user time) 1.806000 s
[TG-TGR ] [info] Test generation summary written: test_summary/tt_result.csv
[U-TW ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-TT-test_
suite.json

```

22. ábra - Output of Transition Tour generation

Test unreliable reset symbols

To be able to generate the HSI, H-method and N-Switch Coverage test cases, I had to add resets to my model.¹⁴

```

E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -o conversion -m add_unreliable_reset -f converted/pinterest_fsm.json
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[U-CTRL ] [info] Unreliable reset succesfully enabled for model: converted/pinterest_fsm.json

```

23. ábra - Output of the generation of reset symbols

Then I generated a test to see, if the reset transitions are reliable.

```

E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -m RCS -f converted/pinterest_fsm_resets.json
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[U-LMI ] [info] Model Name: pin_saving_and_board_creating_at_pinterest-transformed, Type: FSM, Reset: Yes (UN
RELIABLE/FLAG)
[U-CTRL ] [info] Running ResetCheckSequence test generation
[TG-TGR ] [info] Test generation summary written: test_summary/rcs_result.csv
[U-TW ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-RCS-test_
suite.json

```

24. ábra - Output of generation of reset symbol test suite

Harmonized State Identifiers (improved version)

„The Harmonized State Identifiers (HSI) state verification method can be used to create a structured test suite for reduced, deterministic, strongly connected FSMs with reliable reset capability.”

„...it guarantees to find all output and transfer faults...”

„...denotes the possible extra states in implementation compared to the specification...”

¹⁴ I copied this file, and named it „pinterest_fsm_resets.json”

I used the deterministic and complete FSM with the generated reliable reset transitions (pinterest_fsm_reset.json) as source to generate HSI test cases.

I generated an n-complete test suite (there aren't any extra states in the implementation):

```
E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -m HSI -f converted/pinterest_fsm_resets.json
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[U-LMI ] [info] Model Name: pin_saving_and_board_creating_at_pinterest-transformed, Type: FSM, Reset: Yes (UNRELIABLE/FLAG)
[U-CTRL ] [info] Running Harmonized State Identifiers test generation, extra states: 0
[TG-TGR ] [warning] The reset symbols of the given model are unreliable. The already generated Reset Check Sequence should be applied before use the test suite of this test generation algorithm: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-RCS-test_suite.json
[TGC-SPG] [warning] The model is not reduced!
[TGC-SPG] [warning] The model has a state pair that cannot be separated nor merged!
```

25. ábra - Warning messages of HSI generation

```
[TG-HSI ] [info] Verifying state profile;boardQuantity=0;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=false
[TG-HSI ] [info] Verifying state home_page;boardQuantity=0;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=false
[TG-HSI ] [info] Inserting and checking the remaining transitions...
[TG-HSI ] [info] 2nd stage: Test remaining transitions
[TG-HSI ] [info] Generating test suite...
[TG-HSI ] [info] The length of the test suite is 17232
[TG-HSI ] [info] Finished computation at 2024-04-14 17:50:29.5390368
elapsed time: (real time) 11.424 s
elapsed time: (user time) 11.424000 s
[TG-TGR ] [info] Test generation summary written: test_summary/hsi_result.csv
[U-TW ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-HSI-extra_states-0-test_suite.json
```

26. ábra - Output of n-complete HSI generation

And one with 2 extra states:

```
[TG-HSI ] [info] Verifying state board_page;boardQuantity=0;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=false
[TG-HSI ] [info] Verifying state profile;boardQuantity=0;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=false
[TG-HSI ] [info] Verifying state home_page;boardQuantity=0;createBoardModalWasOpenedFromProfile=false;isSelectedPinSaved=false
[TG-HSI ] [info] Inserting and checking the remaining transitions...
[TG-HSI ] [info] 2nd stage: Test remaining transitions
[TG-HSI ] [info] Generating test suite...
[TG-HSI ] [info] The length of the test suite is 295373
[TG-HSI ] [info] Finished computation at 2024-04-14 18:03:15.3561736
elapsed time: (real time) 35.0061 s
elapsed time: (user time) 35.006000 s
[TG-TGR ] [info] Test generation summary written: test_summary/hsi_result.csv
[U-TW ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-HSI-extra_states-2-test_suite.json
```

27. ábra - Output of the generation of HSI with 2 extra states

H-method

„The H-method is an improvement of the HSI-method. It creates a test suite for reduced, deterministic and strongly connected FSMs with reliable reset capacity, but the length of the test suite can be considerably shorter than the one generated by the HSI-method. The resulting test suite is μ -complete, which can distinguish every non-conforming implementation having up to μ states”

I generated an n-complete test suite (there aren't any extra states in the implementation) using the node order number 4:


```
[TGC-SCS] [info] q(n295): [ clickOnUnsavedPin clickOnBoardNamePlusArrow clickOnCreateBoard enterNonExistingNameAndFillInTheForm clickOnUnsavedPin clickOnBoardNamePlusArrow clickOnCreateBoard enterNonExistingNameAndFillInTheForm clickOnUnsavedPin clickOnBoardNamePlusArrow clickOnCreateBoard enterNonExistingNameAndFillInTheForm clickOnProfileIcon clickOnPlusIcon clickOnBoard enterNonExistingNameAndClickOnCreate clickOnDoneBtn clickOnOptions clickOnEdit clickonDelete ]
[TG-H ] [info] Collecting input and output symbols...
[TG-H ] [info] Creating testing tree...
[TG-H ] [info] Stage 2: appending testing tree with separating sequences of (SC-SC) pairs...
[TG-H ] [info] Stage 3: appending testing tree with separating sequences of (SC-extra) pairs...
[TG-H ] [info] Stage 4: appending testing tree with separating sequences of (extra-extra) pairs...
[TG-H ] [info] Generating test suite...
[TG-H ] [info] The length of the test suite is 15563
[TG-H ] [info] Finished computation at 2024-04-14 18:13:29.8224928
elapsed time: (real time) 49.6526 s
elapsed time: (user time) 49.653000 s
[TG-TGR ] [info] Test generation summary written: test_summary/h_result.csv
[U-TW ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-H-node_order-4-extra_states-0-test_suite.json
```

28. ábra - Output of H-method test generation

And one with 2 extra states:

```
[TGC-SCS] [info] q(n295): [ clickOnUnsavedPin clickOnBoardNamePlusArrow clickOnCreateBoard enterNonExistingNameAndFillInTheForm clickOnUnsavedPin clickOnBoardNamePlusArrow clickOnCreateBoard enterNonExistingNameAndFillInTheForm clickOnUnsavedPin clickOnBoardNamePlusArrow clickOnCreateBoard enterNonExistingNameAndFillInTheForm clickOnUnsavedPin clickOnBoardNamePlusArrow clickOnCreateBoard enterNonExistingNameAndFillInTheForm clickOnProfileIcon clickOnPlusIcon clickOnBoard enterNonExistingNameAndClickOnCreate clickOnDoneBtn clickOnOptions clickOnEdit clickonDelete ]
[TG-H ] [info] Collecting input and output symbols...
[TG-H ] [info] Creating testing tree...
[TG-H ] [info] Stage 2: appending testing tree with separating sequences of (SC-SC) pairs...
[TG-H ] [info] Stage 3: appending testing tree with separating sequences of (SC-extra) pairs...
[TG-H ] [info] Stage 4: appending testing tree with separating sequences of (extra-extra) pairs...
[TG-H ] [info] Generating test suite...
[TG-H ] [info] The length of the test suite is 266266
[TG-H ] [info] Finished computation at 2024-04-14 18:38:31.9378493
elapsed time: (real time) 1011.69 s
elapsed time: (user time) 1011.695000 s
[TG-TGR ] [info] Test generation summary written: test_summary/h_result.csv
[U-TW ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-H-node_order-4-extra_states-2-test_suite.json
```

29. ábra - Output of H-method test generation with 2 extra states

N-Switch Coverage

„The N-switch-coverage test generation method covers all topologically possible, consecutive N+1 transitions.”

I generated 1-switch-coverage test sequence:

```
E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -m NS -f converted/pinterest_fsm_resets.json
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[U-LMI ] [info] Model Name: pin_saving_and_board_creating_at_pinterest-transformed, Type: FSM, Reset: Yes (UNRELIABLE/FLAG)
[U-CTRL ] [info] Running NSwitchCoverage test generation
[TG-1-SC] [info] Start of building map of paths to traverse
[TG-1-SC] [info] Starting test sequence generation from the map of consecutive 2 long transitions
[TG-1-SC] [info] Traversal complete
[TG-1-SC] [info] Result:
length of test sequence: 7832,
number of paths: 3781
[TG-1-SC] [info] Finished computation at 2024-04-14 18:44:57.0401827
elapsed time: (real time) 205.306 s
elapsed time: (user time) 205.307000 s
[TG-TGR ] [info] Test generation summary written: test_summary/ns_result.csv
[U-TW ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-NS-1-test_suite.json
```

30. ábra –Output of 1-Switch Coverage test generation


```

E:\Programs\ModelTestRelax\MTR-3-4-0>MTR -m NS --ns_value=2 --ns_map -f converted/pinterest_fsm_resets.json
[U-CTRL ] [info] Version: 3.4.0 R4: Belted kingfisher, profile: DEFAULT, verbosity: 3, debug mode: off
[U-LMI  ] [info] Model Name: pin_saving_and_board_creating_at_pinterest-transformed, Type: FSM, Reset: Yes (UNRELIABLE/FLAG)
[U-CTRL ] [info] Running NSwitchCoverage test generation
[TG-2-SC] [info] Start of building map of paths to traverse
[TG-2-SC] [info] Map (paths to traverse) file written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-NS-2-Map_of_transition_sequences.json
[TG-2-SC] [info] Starting test sequence generation from the map of consecutive 3 long transitions
[TG-2-SC] [info] Traversal complete
[TG-2-SC] [info] Result:
    length of test sequence: 34995,
    number of paths: 14635
[TG-2-SC] [info] Finished computation at 2024-04-14 19:17:29.7153501
    elapsed time: (real time) 1460.8 s
    elapsed time: (user time) 1460.803000 s
[TG-TGR ] [info] Test generation summary written: test_summary/ns_result.csv
[U-TW   ] [info] Test suite written: test_suites/pin_saving_and_board_creating_at_pinterest-transformed-NS-2-test_suite.json

```

31. ábra - Output of 2-Switch Coverage test generation

Results of the generations

method	state_count	transition	duration_real	test_sequence_length
Random	296	979	1,185800	469226
Random	296	979	0,833483	320303
Random	296	979	0,533258	209443
TT	296	979	1,806070	1498
ResetCheckSequence	296	1276	0,040312	4566

method	state_count	transition	duration_real	test_sequence_length
HSI (0 extra state)	296	979	11,42400	17232
H (0 extra state)	296	979	49,65260	15563
HSI (2 extra state)	296	979	35,00610	295373
H (2 extra state)	296	979	1011,69000	266266
1-Switch	296	979	205,30600	7832
2-Switch	296	979	1460,80000	34995

I attached all my original and generated models, test cases and generation summaries / results to this documentation.