ETH zürich

FS 2021

Software Requirements Specification

for

digital UNO

Version 0.5 approved

Prepared by Achart Marie-Louise; Mazzoleni Alice; Heer Damian; Schaufelberger Andreas; Alija Valon; Gutmann Katharina

Cereal Killers

15.03.2021

Table of Contents

Revision History

Name	Date	Release Description	Version
Felix Friedrich	03/15/21	Template for Software Engineering Course in ETHZ.	0.3
Cereal Killers	03/15/21	First draft to be reviewed	0.4
Cereal Killers	03/18/21	Final version	0.5

1. Introduction

1.1 Purpose

<Identify the product whose software requirements are specified in this document, including the revision or release number. Describe the scope of the product that is covered by this SRS, particularly if this SRS describes only part of the system or a single subsystem.>

1.2 Document Conventions

<Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. For example, state whether priorities for higher-level requirements are assumed to be inherited by detailed requirements, or whether every requirement statement is to have its own priority.>

1.3 Intended Audience and Reading Suggestions

<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers. Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>

1.4 Product Scope

<Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals. Relate the software to corporate goals or business strategies. If a separate vision and scope document is available, refer to it rather than duplicating its contents here.>

1.5 References

<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document. Provide enough information so that the reader could access a copy of each reference, including title, author, version number, date, and source or location.>

2. Overall Description

2.1 Product Perspective

<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the

larger system to the functionality of this software and identify interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful.>

2.2 Product Functions

- Players will be uniquely identified with a name.
- At the beginning the uno game will provide seven cards to each player and lay one card as the discard pile.
- The uno game will display to all the players the most recent card of the discard pile.
- The uno game will display to each user their own set of cards at hand.
- The game will let the players play one after the other in a given order.
- On a player turn, the player can either put down a card on the discard pile or pick a card from the remaining deck.
- Once a player has gotten rid of all his cards he wins the game.
- The remaining players can continue the game until one of the last two players wins.
- Once a player won, he can remain (if he wants) in the game as a spectator but cannot play anymore.
- General Uno Rules.

2.3 User Classes and Characteristics

The set of users of the digital version of the Uno game is the same as the set of users playing with the paper card version. Humans from 7 to 77 years old, with interest in playing a digital version of their favorite game.

2.4 Operating Environment

The digital Uno game will operate with two software components: a client side (implemented with wxWidgets) and a server side, on a UNIX-based system.

Each player will have a client program running (implemented with wxWidgets), that displays the current state of the game and lets the player perform actions. The server program stores the current state of the game and is responsible for coordinating the data between the client programs.

2.5 Design and Implementation Constraints

<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer's organization will be responsible for maintaining the delivered software).>

2.6 User Documentation

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>

2.7 Assumptions and Dependencies

<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>

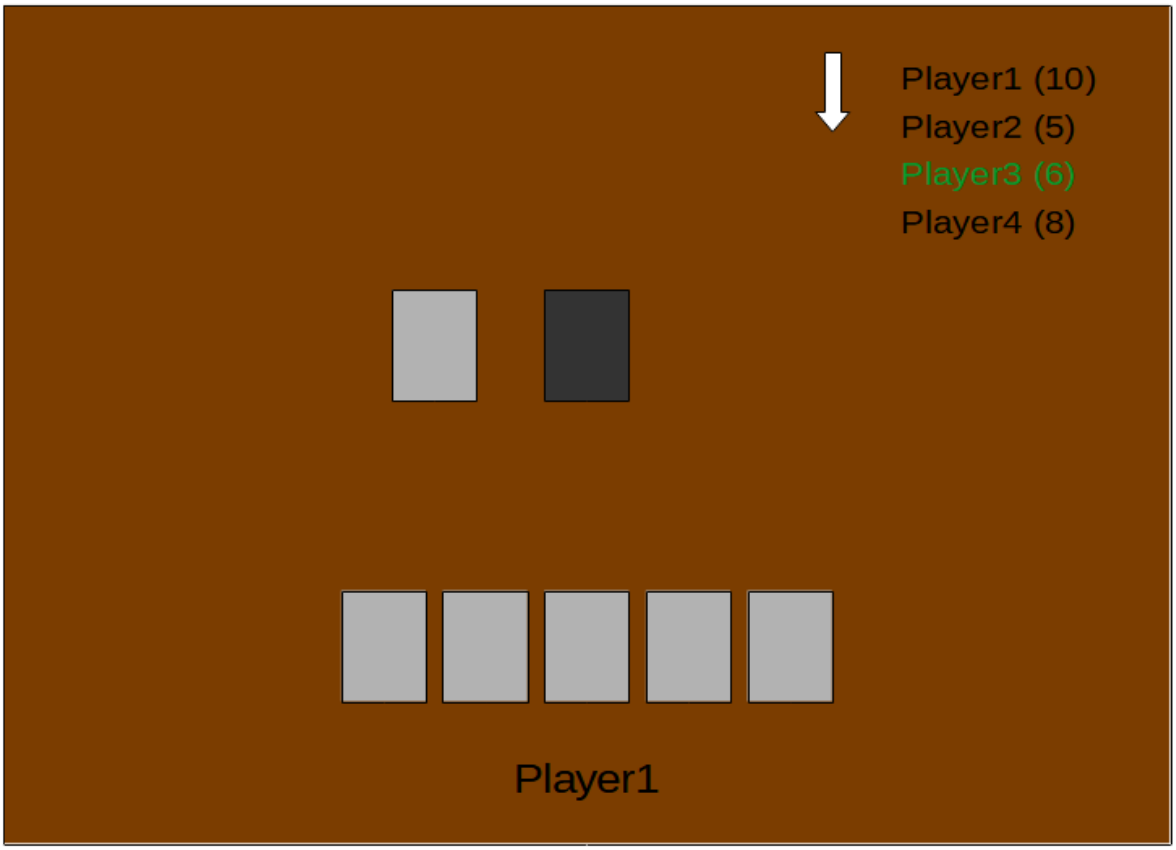
3. External Interfaces and Requirements

3.1 User Interface

The look of the user interface must feel simple and straightforward. There exists a start game and an exit button.

Every player will see its own set of cards, the top card of the discard pile and the pile of cards to be drawn. One can play a card by clicking onto the respective card and draw a card by clicking onto the pile of cards to be drawn. The interface also displays the list of other players with their name and their number of cards at hand, highlighting which user is currently playing (to allow every player to follow the game).

The colour to be matched is also displayed, allowing the behavior of special cards. If an invalid card is played the server will send an error message which appears on screen.



3.2 Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>

3.3 Communications Interfaces

The interaction between server and client will use TCP with wxWidgets different classes, to help with interprocess communication and network programming.

4. System Requirements

4.1 Functional Requirements

4.1.1 FREQ-1: Uniqueness User id

User id must be unique.
User Priority (2/5)
Technical Priority (5/5)

4.1.2 FREQ-2: Number of players

At least two and at most four users need to be signed in to start a game.
User Priority (5/5)
Technical Priority (3/5)

4.1.3 FREQ-3: Graphic User Interface

The system should have a Graphic User Interface.
User Priority (5/5)
Technical Priority (1/5)

4.1.4 FREQ-4: Displaying number of cards

The number of cards each player has should be visible for each player.
User Priority(2/5)
Technical Priority (0/5)

4.1.5 FREQ-5: Displaying user-id

The system should display the user-id of all the players currently playing.
User Priority (3/5)
Technical Priority (0/5)

4.1.6 FREQ-6: Start game

The system should allow a user to start a game by clicking on the Start Game button.
User Priority (5/5)
Technical Priority (4/5)

4.1.7 FREQ-7: Ruleset of Uno

The system uses the common rules of Uno, as to be found on:
https://en.wikipedia.org/wiki/Uno_(card_game)#Official_rules.
No scores are computed at the end. If a player has played its last card, he wins. There is no need to say "Uno". There are no penalties when playing a "Wild Draw 4" card. A player can still play a card after having drawn cards due to a +2 or Wild Draw 4 card.
User Priority (4/5)
Technical Priority (4/5)

4.1.8 FREQ-8: Winning a game

If one player wins, a Popup window naming "Player X wins" appears.
User Priority (2/5)
Technical Priority (0/5)

4.1.9 FREQ-9: Exit game

The system should allow a user to exit the game at any given moment.
User Priority (2/5)
Technical Priority (2/5)

4.1.10 FREQ-10: Play/Draw

The system should allow the user to select with a click the card she/he wants to put down on the discard pile or to draw from the remaining deck by clicking onto the remaining deck.
User Priority (5/5)
Technical Priority (4/5)

4.1.11 FREQ-11: Turn-based game updates

The system should be able to handle turn-based game updates.
User Priority (5/5)
Technical Priority (5/5)

4.1.12 FREQ-12: Display discard pile

The system should allow every user to follow the game, i.e. to see the last card on top of the discard pile.
User Priority (5/5)
Technical Priority (2/5)

4.1.13 FREQ-13: Joining a Game

The system should not allow any additional user to join a game once the game has started.
User Priority (2/5)

Technical Priority (4/5)

4.1.14 FREQ-14: Uno

If one player plays his second to last card a Popup window naming “Uno” appears and disappears shortly afterwards.
User Priority (3/5)
Technical Priority (0/5)

4.1.15 FREQ-15: HIghlighting of current player

The current player is being highlighted.
User Priority (4/5)
Technical Priority (0/5)

4.1.16 FREQ-16: Draw cards

If a player played a “Draw 2 Action” card or a “Wild draw Four” card, the respective cards are automatically dealt to the next player.
User Priority (1/5)
Technical Priority (0/5)

4.1.17 FREQ-17: Timeout

If a player doesn’t play in a given time slot, a card will automatically be handed out to the player and the player misses this turn.
User Priority (4/5)
Technical Priority (0/5)

4.1.18 FREQ-18: Colour selection

If a player plays a +4/wild card, a selection window pops up where the user can select one of the colours (green, red, yellow, blue).
User Priority (4/5)
Technical Priority (3/5)

4.1.19 FREQ-19: Admissibility

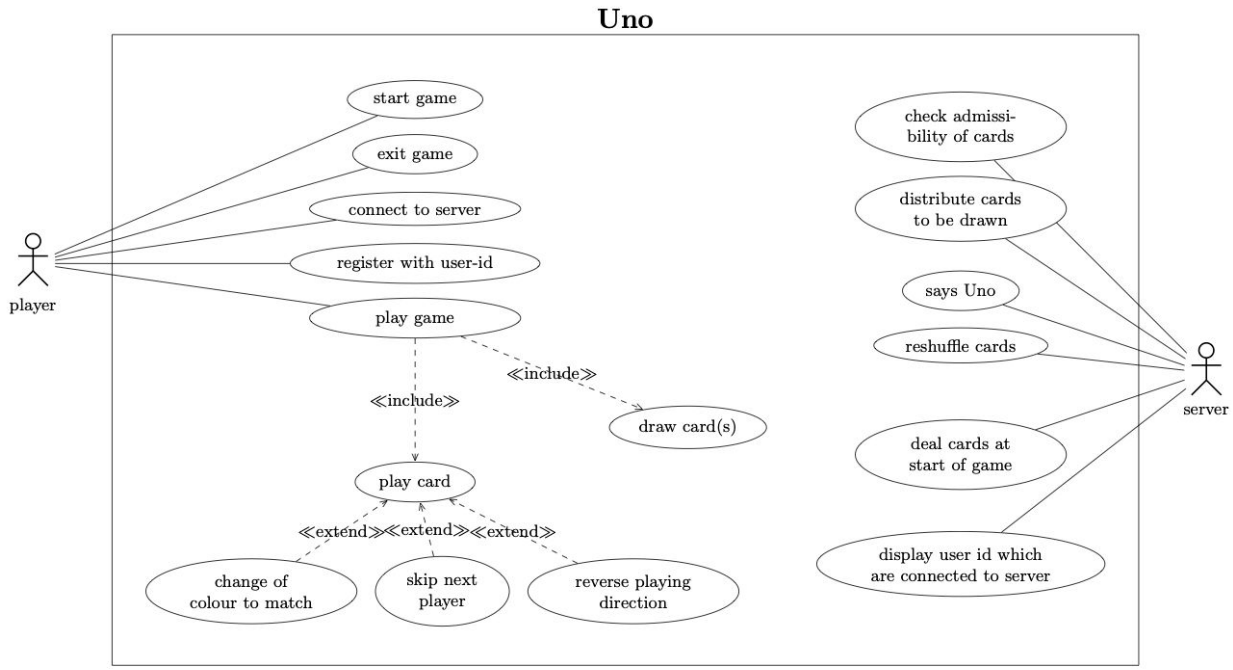
If a player plays an invalid card the server notifies the player by displaying an error along with some rule for reference.
User Priority (3/5)
Technical Priority (0/5)

4.1.20 FREQ-XX: Placeholder

User Priority (x/5)
Technical Priority (x/5)

5. System Scenarios

5.1 Use-case Diagrams



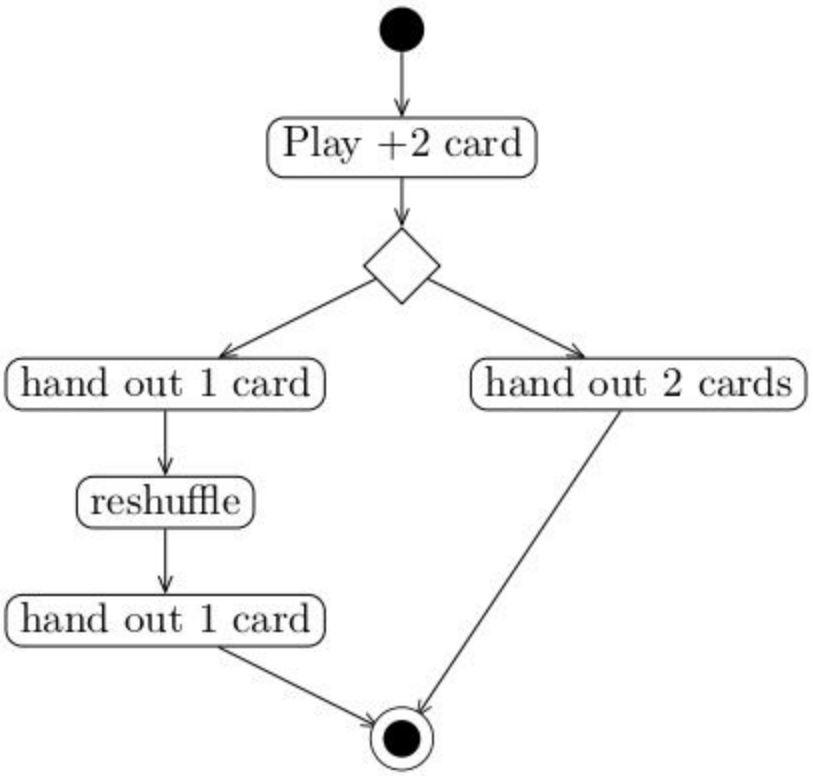
5.2 Scenarios

5.2.1 SCN-1: Deck Reshuffle

FREQ reference	*3, 7, 10, 16, 17, 15*
NFREQ reference	*2*
Short Description:	*The user will draw 2 cards. After drawing the first card the deck is empty. Reshuffling the cards of the discard pile and then drawing another card is required.*
Activation action:	*Automatic when given a card from Deck. (Here specifically from a +2 card)*
Precondition:	Only 1 card on the deck pile. Player 1's turn.

Basic flow: Reshuffling		
Step	**User action**	**System response**
1	Player 1 plays a +2 card.	Analyzes if the card is valid and what to do.
2		*Tries to hand out 2 cards to the next player 2.*
3		*Recognises that the deck pile is empty after 1 card*

		and proceeds to reshuffle the discarded pile to use it as the deck pile.
4		*Starts player 2s turn. Highlights 2 players user-id*
Post-condition:	*"full" pile deck of cards. Only top card remains on the discard pile.* Player 2's turn.	

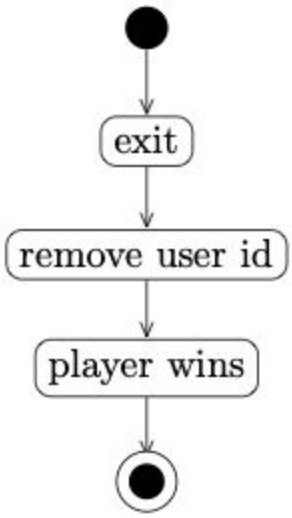


5.2.2 SCN-2: Only two Players are in the game and one leaves

FREQ reference	*2,9,8*
NFREQ reference	*2*
Short Description:	*Player leaves the game, when there are only two players in the game. The remaining person wins the game.*
Activation action:	*Player leaves the game before the end*
Precondition:	*There are only two players in the game*

Basic flow: Only two players are in the game and one leaves		
Step	**User action**	**System response**
1	*user leaves game*	*"User-id wins" pops up.*
2		*Remaining user cannot continue playing.*
Post-condition:	*Game is ended. The player who was last in the game wins.*	

Scenario Diagram for SCN-2 Only two players are in the game and one leaves

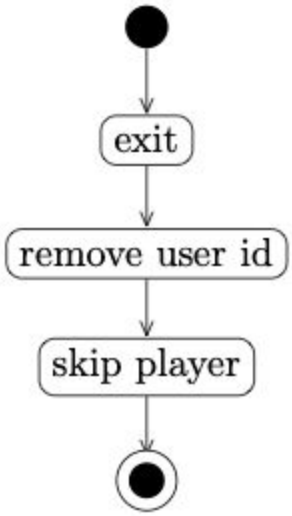


5.2.3 SCN-3: One player exits but more than one player are still in the game

FREQ reference	*5, 9*
NFREQ reference	*2*
Short Description:	*A player exits. The other players will continue the game.*
Activation action:	*One player leaves the game.*
Precondition:	*There are more than 2 players in the game and the game is ongoing.*

Basic flow: One player exits but more than one player are still in the game		
Step	**User action**	**System response**
1	*Player leaves the game.*	*User id of the player which has left isn't displayed anymore.*
2		*The next player takes the turn.*
3		*The player which has left is ignored for the remaining game phase. The remaining cards are put into the discard pile.*
Post-condition:	*The game is continued with the remaining players, while ignoring the position of the player that left and the cards of the player in the discard pile.*	

Scenario Diagram for SCN-3 One player exits but more than one player are still in the game



5.2.4 SCN-4: Player wins game in a game in which more than 2 players are participating

FREQ reference	*8, 9, 15*
NFREQ reference	*2*
Short Description:	*A player wins the game, given he has no more cards left. The remaining Players can continue to play the game and fight for the 2nd, 3d or 4th place.*
Activation action:	*Player has no more cards*

Precondition:	*Player has only one card left.*

Basic flow: Player wins game in a game in which more than 2 players are participating		
Step	**User action**	**System response**
1	*plays last card*	*"user id wins" pops up*
2		*system changes players turn according to played card and highlights its user-id.*
3	*remains in game*	
4		*skips player who has won*
Post-condition:	*Player who has no more cards is not allowed to play anymore.*	

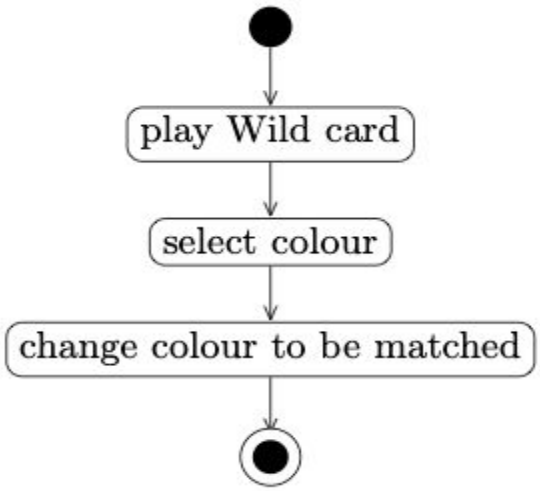
Scenario Diagram for SCN-4 Player wins game in a game in which more than 2 players are participating

5.2.5 SCN-5: Player plays Wild card

FREQ reference	*7, 18*
NFREQ reference	*2*
Short Description:	*A Player plays a Wild card to choose the color. Window pops up where he can choose the color of the card, which will then be applied.*
Activation action:	*playing a Wild card/Wild 4 draw card*
Precondition:	*The played card is admissible.*

Basic flow: Player plays Wild card		
Step	**User action**	**System response**
1	*User plays Wild card*	*Evaluation/admissibility of card*
		Window pops up where one can click on one of the 4 colours (red, yellow, green, blue)
2	*Player chooses one color*	
3		*the colour to be matched will be displayed on the top of the screen*
4		*switch turn*
Post-condition:	*The colour to be matched changed according to the selected colour. Other players turn.*	

Scenario Diagram for SCN-5 Player plays Wild card



6. System Constraints

6.1 Important Nonfunctional Requirements

6.1.1 NFREQ-1: User Interface

The system should be easy to learn how to use for an inexperienced user.
User Priority(5/5)
Technical Priority(0/5)

6.1.2 NFREQ-2: Communication

Server <-> Client communication is implemented with TCP.
User Priority(1/5)
Technical Priority(4/5)

6.1.3 NFREQ-3: Response Time

The system should have a delay of less than 5 seconds.
User Priority(3/5)
Technical Priority(2/5)

6.1.4 NFREQ-4: Freezing time

The system should freeze or close prematurely in less than 10 %.
User Priority(3/5)
Technical Priority(2/5)

7. Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>