# Flashmap server Documentation

***Release 1.0***

**M.C. van den Enk**

**Mar 06, 2017**

# CONTENTS

Contents:

**show-inheritance**

**class** concept_map.**ConceptMap**(**args*, ***values*)

A class representing a concept map

**Parameters**

- **nodes** (*ListField(Node)*) – a list of nodes (by default all existing node documents)
- **edges** (*ListField(Edge)*) – a list of edges (by default all existing edge documents)

**get_partial_map**(*edge*)

Returns a concept map containing only the parent and sibling edges together with the referred nodes

**Parameters** **edge** (Edge) – The input edge

**Returns** A concept map containing parent and sibling edges of edge together with the referred nodes

**Return type** *ConceptMap*

---

**Todo**

Implementation

---

CHAPTER

# ONE

# CONSUMER MODULE

**show-inheritance**

**class** consumer.**Consumer**

This is the class from which the program is controlled. It can be used together with the *handler* module in order to communicate with an external client over a websocket

**Parameters**

- **concept_map** (ConceptMap) – The concept map object containing references to nodes and edges
- **SOURCES** (*list(str)*) – All of the sources referenced to in the edges of the concept map
- **user** (User) – The active user

**authenticate**(*name*)

A function to either return an existing *user.User* or a new *user.User* based on the given name

**Parameters** **name** (*str*) – The username

**Returns** The user with this username

**Return type** *User*

**consumer**(*keyword*, *data*)

Pass data to the function corresponding to the provided keyword for the provided user

**Parameters**

- **keyword** (*str*) – the keyword for which function to use
- **data** (*dict(str, str or dict)*) – the data necessary for executing the function

**Returns** Contains the keyword and data to send over a websocket to a client

**Return type** dict(str, str or dict)

CHAPTER

TWO

# EDGE MODULE

**show-inheritance**

**class** edge.**Edge**(**args*, ***values*)

A class representing an edge from a concept map :param from_node: The parent node of the edge :type from_node: Node :param to_node: The child node of the edge :type to_node: Node :param label: A label describing the relation between from_node and to_node :type label: StringField :param source: The source where this edge is described (e.g. paragraph 13.2 from Laagland) :type source: StringField

CHAPTER

# THREE

# FLASHCARD MODULE

**show-inheritance**

**class** flashcard.**Flashcard**(**args*, ***values*)

A class representing a flashcard :param question: The question on the front side of the flashcard :type question: StringField :param answer: The answer on the back side of the flashcard :type answer: StringField :param sources: The sources where this flashcard are described (e.g. paragraph 13.2 of Laagland) :type sources: ListField(StringField) :param response_model: A list consisting of parts of valid responses to the question (for the test matrix) :type response_model: ListField(StringField)

CHAPTER

# FOUR

# FLASHCARD_INSTANCE MODULE

**show-inheritance**

**class** flashcard_instance.**FlashcardInstance**(**args*, ***kwargs*)
A class for storing responses from the flashmap system

**Parameters** **reference** (Edge) – The edge to which this instance refers

CHAPTER

FIVE

# FLASHMAP_INSTANCE MODULE

**show-inheritance**

**class** flashmap_instance.**FlashmapInstance**(**args*, ***kwargs*)
A class for storing responses from the flashmap system

**Parameters** **reference** (Edge) – The edge from the concept map to which this instance refers to

CHAPTER

SIX

# HANDLER MODULE

**show-inheritance**

handler.**handler**(*websocket*, *path*)

Initiate an asyncio thread which receives messages from a client, parse the json file to an object, pass them to consumer() and send the result back to the client

**Parameters**

- **websocket** (*Websocket*) – the websocket being used for receiving and sending messages to a client
- **path** (*String*) – the IP address used to host the websocket

CHAPTER

SEVEN

# INSTANCE MODULE

**show-inheritance**

**class** instance.**Instance**(**args*, ***kwargs*)

A class describing a general flash instance, which can either be a FlashmapInstance or a FlashcardInstance

**Parameters**

- **responses** (*ListField(EmbeddedDocumentField(Response))*) – A list of responses provided to this instance (an empty list by default)
- **reference** (*GenericReferenceField*) – A reference to either an edge in a concept map or a flashcard (defined within the subclass)
- **due_date** (*DateTimeField*) – The date this instance is due for repetition

**schedule**()

Reschedules this instance for review based on the previous responses .. todo:: Implementation

CHAPTER

EIGHT

# LOGENTRY MODULE

**show-inheritance**

**class** logentry.**LogEntry**(**args*, ***values*)

An object representing a incoming or outgoing network message

**Parameters**

- **user** (User) – The user which was involved with this network message
- **keyword** (*StringField*) – The network keyword
- **data** (*DictField*) – The dictionary containing the necessary data
- **timestamp** (*DateField*) – The time that this message was received or transmitted

CHAPTER

# NINE

# NODE MODULE

**class** node.**Node**(**args*, ***values*)

Bases: mongoengine.document.Document

A class for representing nodes in the concept map

**Variables** **label** – The label appearing within the node

CHAPTER

# TEN

# QUESTIONNAIRE MODULE

**show-inheritance**

**class** questionnaire.**Questionnaire**(*pu_items*, *peou_items*, ***data*)
A class representing a stored questionnaire for a user

**Variables** **perceived_ease_of_use_items** – Responses to the perceived ease of use item from TAM

**Parameters**

- **good** (*StringField*) – A description of what was good about the software according to the user
- **can_be_improved** (*StringField*) – A description of what could be improved according to the user

**append_answer**(*item*, *phrasing*, *answer*)
Appends an answer to an item within the questionnaire

**Parameters**

- **item** (QuestionnaireItem) – The item to which the answer refers
- **phrasing** (*boolean*) – Whether the item is positively (True) phrased or negatively (False)
- **answer** (*string*) – The answer to be appended

**perceived_usefulness_items**
Responses to the perceived usefulness items from TAM :type: list(QuestionnaireResponses)

CHAPTER

# ELEVEN

# QUESTIONNAIRE_ITEM MODULE

**show-inheritance**

**class** questionnaire_item.**QuestionnaireItem**(**args*, ***values*)

A class representing a single item on the questionnaire

**Parameters**

- **usefullness** – Defines whether the item is part of the perceived usefulness items (True) or of the perceived ease of use items (False)
- **positive_phrasing** (*StringField*) – The version of this item which is positively phrased
- **negative_phrasing** (*StringField*) – The version of this item which is negatively phrased

CHAPTER

# TWELVE

# QUESTIONNAIRE_RESPONSE MODULE

**show-inheritance**

**class** questionnaire_response.**QuestionnaireResponse**(**args*, ***kwargs*)
A class for storing singular responses to questionnaire items

**Parameters**

- **questionnaire_item** (QuestionnaireItem) – The questionnaire item to which this answer refers
- **answer** (*IntField*) – The value of the likert-scale rating the user gave to this item (ranges from -2 to 2)
- **phrasing** (*BooleanField*) – Whether this answer refers to the positively (True) or the negatively (False) phrased version of the questionnaire_item

CHAPTER

# THIRTEEN

# RESPONSE MODULE

**show-inheritance**

**class** response.**Response**(**args*, ***kwargs*)

A class representing a singular response to an Instance.

**Parameters**

- **start** (*DateTimeField*) – The moment the parent Instance was sent to the client
- **end** (*DateTimeField*) – The moment the answer from the client was received
- **correct** (*BooleanField*) – Whether the answer to the Instance was correct (True) or incorrect (False)

CHAPTER

# FOURTEEN

# SESSION MODULE

**show-inheritance**

**class** session.**Session**(**args*, ***kwargs*)

A class representing a session the user was logged in

**Parameters**

- **start** (*DateTimeField*) – The time that the user logged in
- **end** (*DateTimeField*) – The time that the user logged out
- **source_prompted** (*BooleanField*) – Whether the user was asked to have read a certain source from SOURCES
- **browser** (*StringField*) – The type of browser used to log in

CHAPTER

# FIFTEEN

# TEST MODULE

**show-inheritance**

**class** test.**Test**(*flashcards*, *items*, *prev_flashcards=[]*, *prev_items=[]*, ***data*)

A class representing a pre- or posttest the user filled in

**Parameters**

- **test_flashcard_responses** (TestFlashcardResponse) – A list of responses to the flashcard questions on the test
- **test_item_responses** (TestItemResponse) – A list of responses to the item questions on the test

**generate_test**(*items*, *prev_items*)

A method for taking five random items in a random order from the provided list of items without the items in the previous items

**Parameters**

- **items** (*list(Flashcard) or list(TestItem)*) – The complete list of items
- **prev_items** (*list(Flashcard) or list(TestItem)*) – The list of items to be excluded from the result

**Result** A sample of five items from items not included in prev_items

**Return type** list(FlashcardResponse) or list(*TestItemResponse*)

CHAPTER

# SIXTEEN

# TEST_FLASHCARD_RESPONSE MODULE

**show-inheritance**

**class** test_flashcard_response.**TestFlashcardResponse**(**args*, ***kwargs*)
An answer for a flashcard item within a pre- or posttest

**Parameters**

- **answer** (*StringField*) – The answer provided by the user
- **flashcard** (*StringField*) – The flashcard to which this response refers to

CHAPTER

SEVENTEEN

# TEST_ITEM MODULE

**show-inheritance**

**class** test_item.**TestItem**(**args*, ***values*)

A class representing an item from a pre- or posttest

**Parameters**

- **question** (*STringField*) – The question for this item
- **sources** (*ListField(StringField)*) – A list of sources relevant to this question
- **response_model** (*ListField(StringField)*) – A list of the parts of a valid answer used for the test matrix

CHAPTER

# EIGHTEEN

# TEST_ITEM_RESPONSE MODULE

**show-inheritance**

**class** test_item_response.**TestItemResponse**(**args*, ***values*)

A class representing singular answers to test items

**Parameters**

- **answer** (*StringField*) – The answer to item provided by the user
- **item** (TestItem) – The specific item this response refers to

CHAPTER

# NINETEEN

# USER MODULE

**show-inheritance**

**class** user.**User**(**args*, ***values*)
A class representing a user

**Parameters**

- **flashmap_condition** (*BooleanField*) – Whether the user uses the flashmap system (True) or the flashcard system (False)
- **birthdate** (*DateTimeField*) – The birthdate of the user
- **read_sources** (*ListField(StringField)*) – A list of read sources by the user
- **gender** (*StringField*) – The gender of the user (can be either 'male', 'female', or 'other')
- **code** (*StringField*) – The code from the user's informed consent form
- **tests** (*ListField(Test)*) – The pre- and posttest
- **questionnaire** (Questionnaire) – The questionnaire
- **instances** – A list of instances storing the flashmap/flashcard data for the user
- **sessions** (Session) – A list of past sessions for this user

**append_questionnaire**(*responses*, *good*, *can_be_improved*, *email*)
A method for appending a questionnairy to the user given responses

**Parameters**

- **responses** (*dict*) – A list of dict objects containing a QuestionnaireItem (key = 'item') and an answer (key = 'answer')
- **good** (*string*) – A description of what was good about the software according to the user
- **can_be_improved** (*string*) – A description of what can be improved about the software according to the user
- **email** (*string*) – The email address of the user

..todo:: implementation

**append_test**(*flashcard_responses*, *item_responses*)
A method for appending a test to the user given flashcard and item responses

**Parameters**

- **flashcard_responses** (*dict*) – A list of dict objects containing a `Flashcard` (key = 'card') and an answer (key = 'answer')
- **item_responses** (*dict*) – A list of dict objects containing a `TestItem` (key = 'item') and an answer (key = 'answer')

..todo:: fix

**create_questionnaire**(*items*)
A method for creating a new questionnaire

**Parameters** **items** (*list(QuestionnaireItem)*) – A list of questionnaire items

..todo:: implementation

**create_test**(*flashcards*, *items*)
A method for creating a new test with unique questions

**Parameters**

- **flashcards** (*list(Flashcard)*) – A list of flashcards from the database
- **items** (*list(TestItem)*) – A list of items from the database

**set_descriptives**(*birthdate*, *gender*, *code*)
A method for setting the descriptives of the user

**Parameters**

- **birthdate** (*DateTime*) – The provided birthdate of the user
- **gender** (*string*) – The gender of the user (can be either 'male', 'female', or 'other')
- **code** (*string*) – The code from the informed consent form

CHAPTER

TWENTY

# INDICES AND TABLES

- genindex * modindex
  - search

# PYTHON MODULE INDEX

# INDEX