

Inhoud

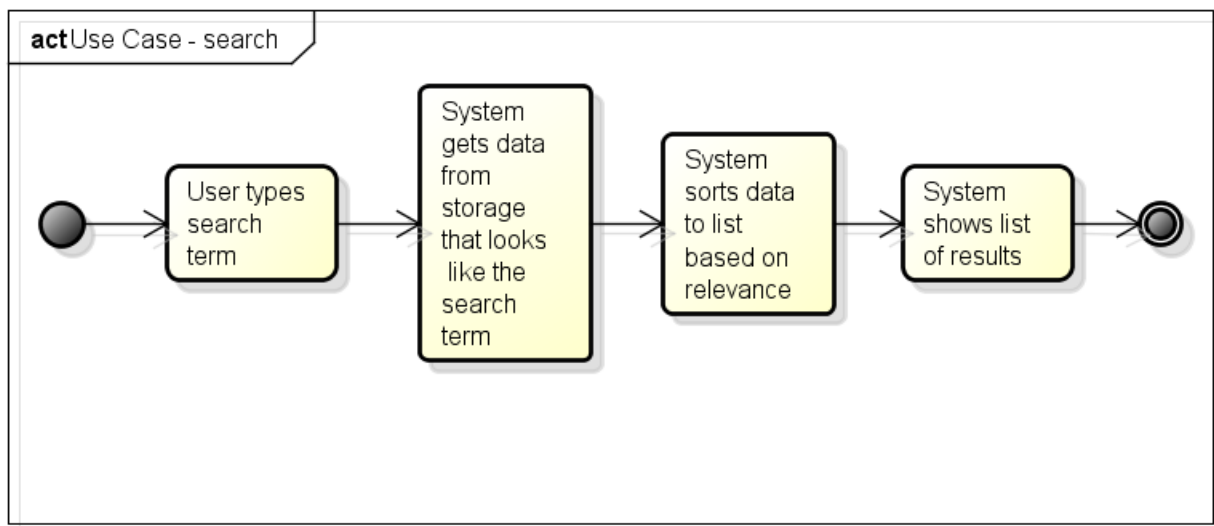
Use Case 1: Search	2
Activity diagram	2
Sequence diagrams	2
Use Case 2: Crawler	3
Activity diagram	3
Use Case 3: Parsen	4
Activity diagram	5
Sequence diagrams	5
Use case 4: Master	6

Use Case 1: Search

The user goes to <http://thywin.com>. The browser then shows a search page. The user types a search term in the search field. The server then shows all the results based on the search term.

Primary Actor: User	
Stakeholders: Site owners	
Preconditions:	
Post conditions:	
Main success scenario:	
1. User goes to http://thywin.com	2. Webserver sends page back
3. User types search term in the search field	4. Webserver shows results.
Extensions: (or Alternative flow)	
	[If no results] 4a. Webserver shows message "0 results"

Activity diagram



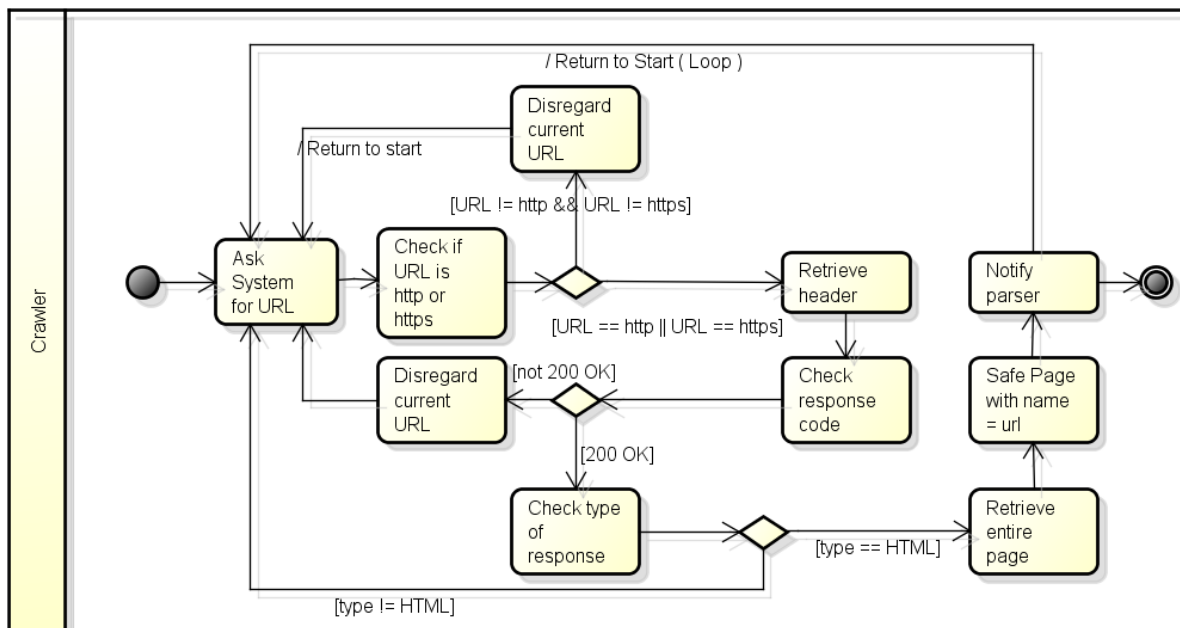
Sequence diagrams

Use Case 2: Crawler

The crawler asks the master for an uri. The master then gets an uri from the uri queue en sends this to the crawler. The crawler then uses wget to get the data from the uri. This data and the uri will be send back to the master.

Primary actor: Crawler	
Stakeholders & interest: Scheduler, Parser	
Preconditions: Connection to the master, connection to the internet	
Main success Scenario	
1. Crawler asks master for a uri to crawl.	2. The master gets an uri from the uri queue.
	3. The master sends an uri to the crawler.
4. Crawler gets the data from the uri using wget.	
5. Crawler sends the uri and the crawldata to the master.	
Extensions: (or Alternative flow)	

Activity diagram

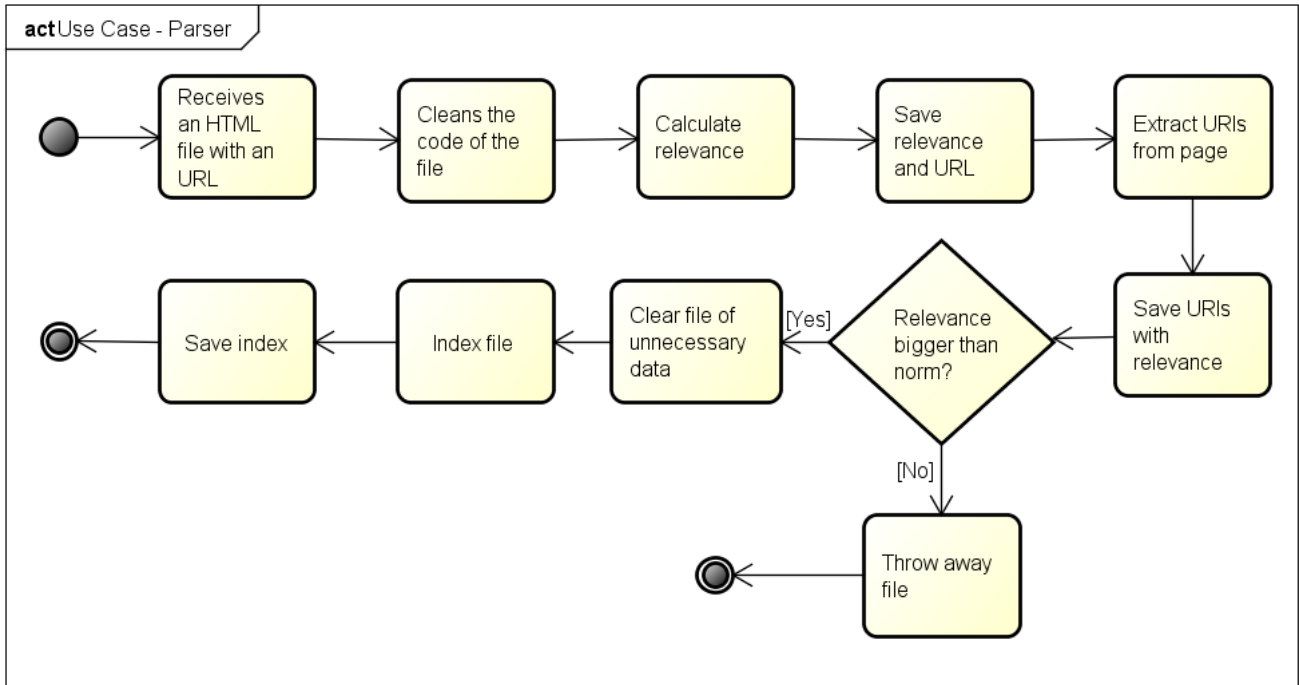


Use Case 3: Parsen

The parser requests an document and the related uri from the master. The parser then extracts all the uri's from the document. The parser then removes the html tags. The parser is now ready to determine the relevance of the document. After this is done the parser sends the found uri's to the master.

Primary Actor: Parser	
Stakeholders: Master, Crawler, Database	
Preconditions: Connection to the database and the crawler, file is HTML format	
Post conditions: File is indexed and the index, URIs and relevance is stored	
Main success scenario:	
1. The parser requests a document from the master.	2. The master send the uri and document to the parser
3. The parser extracts the uri's from the document.	
4. The parser removes the html tags.	
5. The parser determines the relevance of the document.	
6. The parser sends the uri's with the relevance to the master	
Extensions: (or Alternative flow)	

Activity diagram



Sequence diagrams

Use case 4: Master

TBA