News Feed – Booklets

1. Initial Idea

Interactive dashboard managing news articles collected from RSS sources.

Objectives

Enhancing the process of news consulting from the Web with:

- user-friendly RSS reader;
- user-customized selection of sources;
- articles on-the-fly processing;
- easiness in obtaining a pre-attentive comparison of articles from different suppliers.

Distributed Software Application with Containerization

Several components:

- **Front-end Layer**:
  login + landing page + user profile page.
- **API Layer**:
  hosting RESTful Web Server acting as gateway secured with authentication (e.g., OAuth2 with Google Sign-in).
- **Logic Layers**:
  different modules for executing the core business logic. E.g., retrieving RSS feed articles, Natural Language Processing component to summarize articles content. Communication with gateway with the help of RPC.
- **Persistence Layer**.

Each element of the list represents a different Docker container of the system. All the containers are orchestrated using Docker Compose.

Potential Users and Main Use Cases

Potential User: teenager interested in remaining informed about some topics, but who has not the patience for reading entire articles.

Potential User: medium-age person who wants to have an overview of different sources of news for having a pre-attentive comparison of them.

Use Case: user explores the landing page for having an overview of different articles coming from a certain source.

Use Case: user triggers the analysis module to have visualized the full text of a certain article on the landing page.

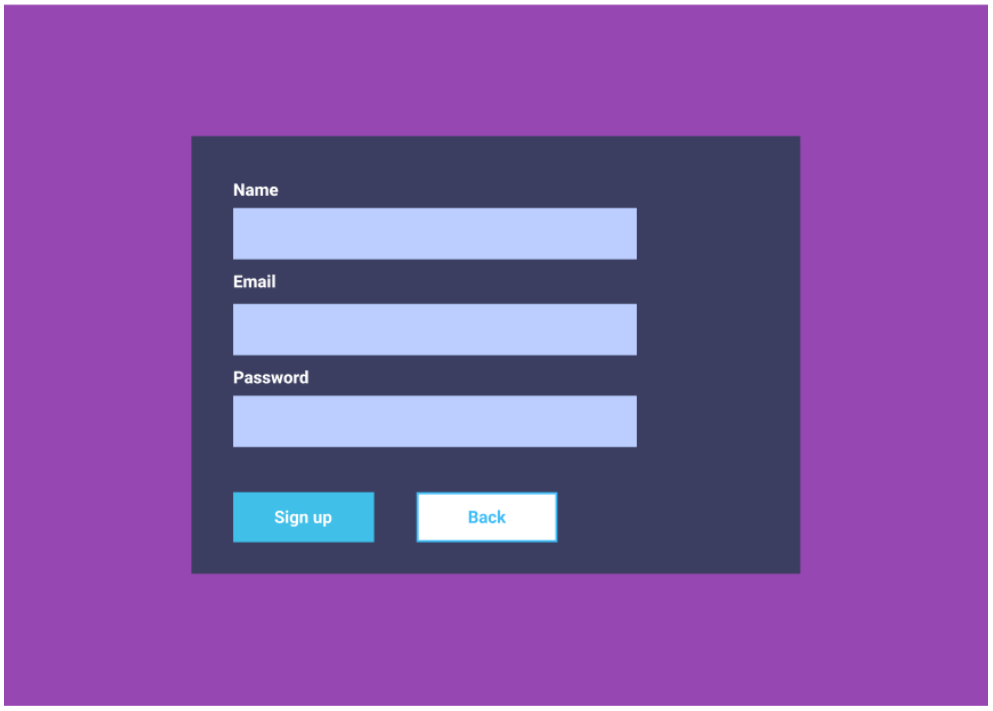
Use Case: user triggers the analysis module to have visualized the summary of a certain article on the landing page.

Use Case: user navigates to its profile page for consulting and managing its personal information.

2. User Stories and Prototypes

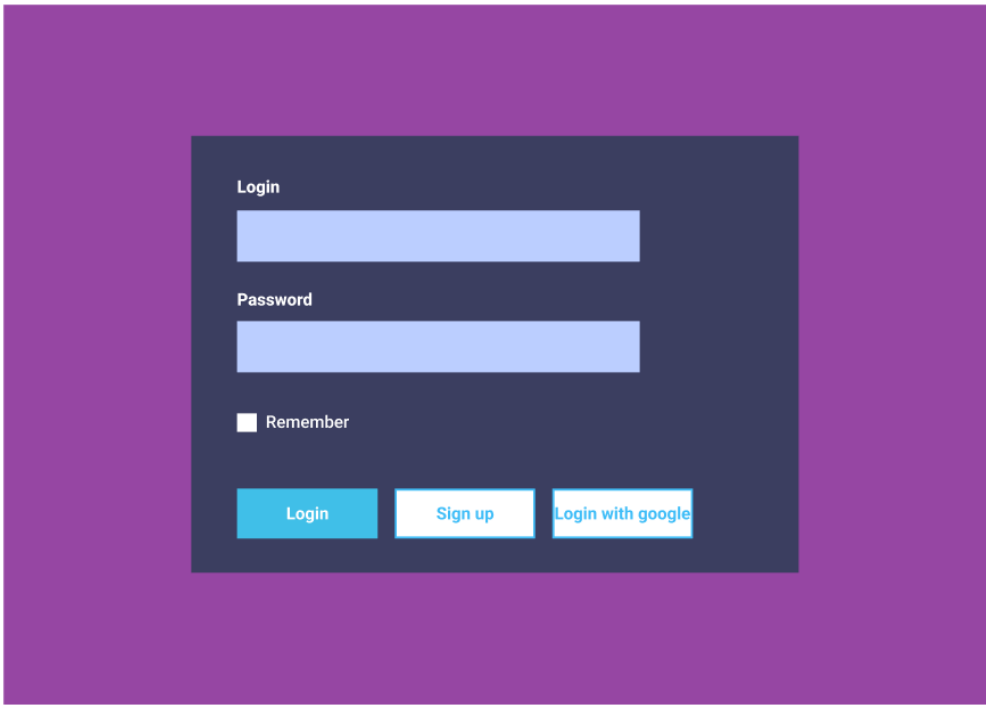
Sign-up

As a	I want to	So that	Date added
User	Subscribe to the system	I can share my personal information in order to let the app create my profile	22/12/2021



Login

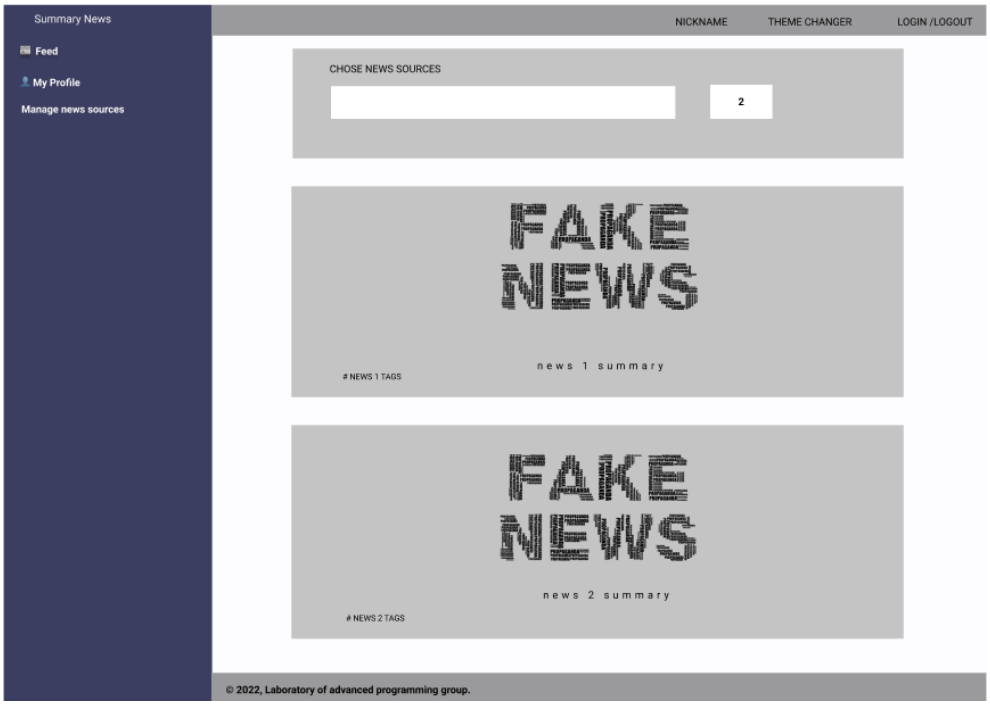
As a	I want to	So that	Date added
User	Log-in into the system (also with google).	I can access the system.	20/12/2021
User	Go to the sign-up page.	I can navigate to the sign-up page, using the sign-up button.	20/12/2021



Main

As a	I want to	So that	Date added
User	Explore the feed page	I can find some news I want to read	16/12/2021
User	Read full text of a specific article	I can select a specific article and read the full text	17/12/2021
User	Obtain short summary of article content	I can obtain and read a short summary of a specific article, instead of reading the full text	17/12/2021
User	Specify an RSS URL to track	I can specify an RSS URL to track	19/12/2021
User	Customize as you prefer the RSS feeds	I can customize the RSS feed articles shown.	06/02/2022
User	Change theme	I can change the theme's color (light or dark), using the Theme Changer button	19/12/2021
User	Log out from the system	I can log out from the system, using the log out button	20/12/2021

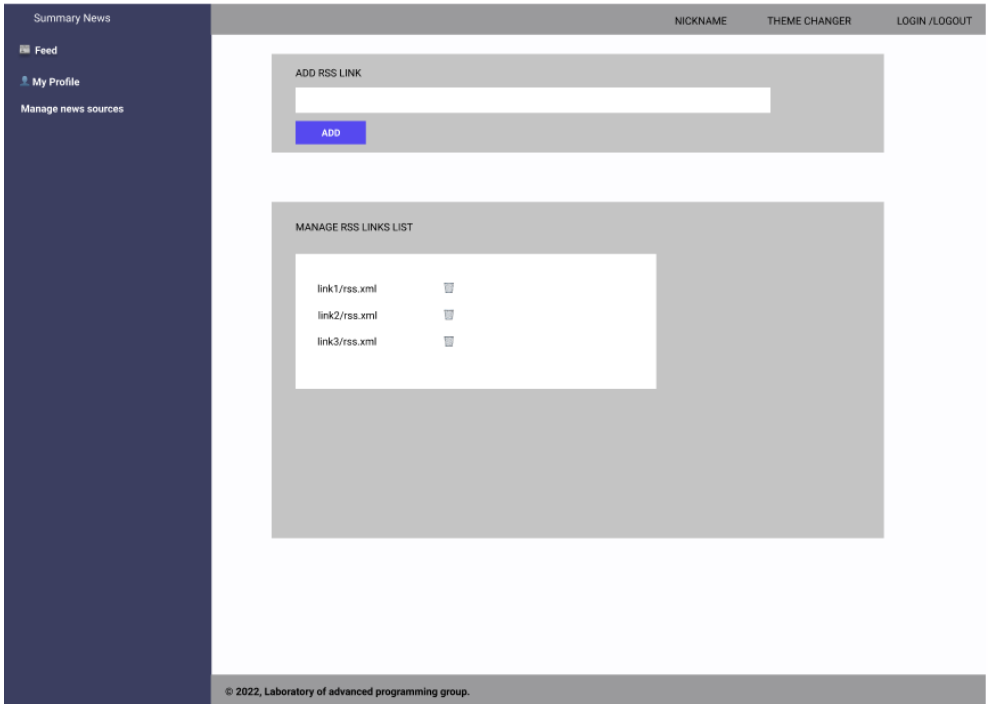
User	Go to the “my profile” page	I can go to the “my profile” page, using a button	16/12/2021
User	Go to the login page	I can go to the login page, using the Login button	16/12/2021
User	Go to the “manage news sources” page	I can go to the “manage news sources” page, using a button	19/12/2021



Manage Sources

As a	I want to	So that	Date added
User	Add RSS links	I can add a specific RSS feed which I’m interested in	21/12/2021
User	Manage RSS links list	I can manage my RSS links list, deleting RSS links that I’m not interested in	22/12/2021
User	Change theme	I can change the theme's color (light or dark), using the Theme Changer button	19/12/2021
User	Go to the Feed page	I can go to the Feed page, using the Feed button	16/12/2021

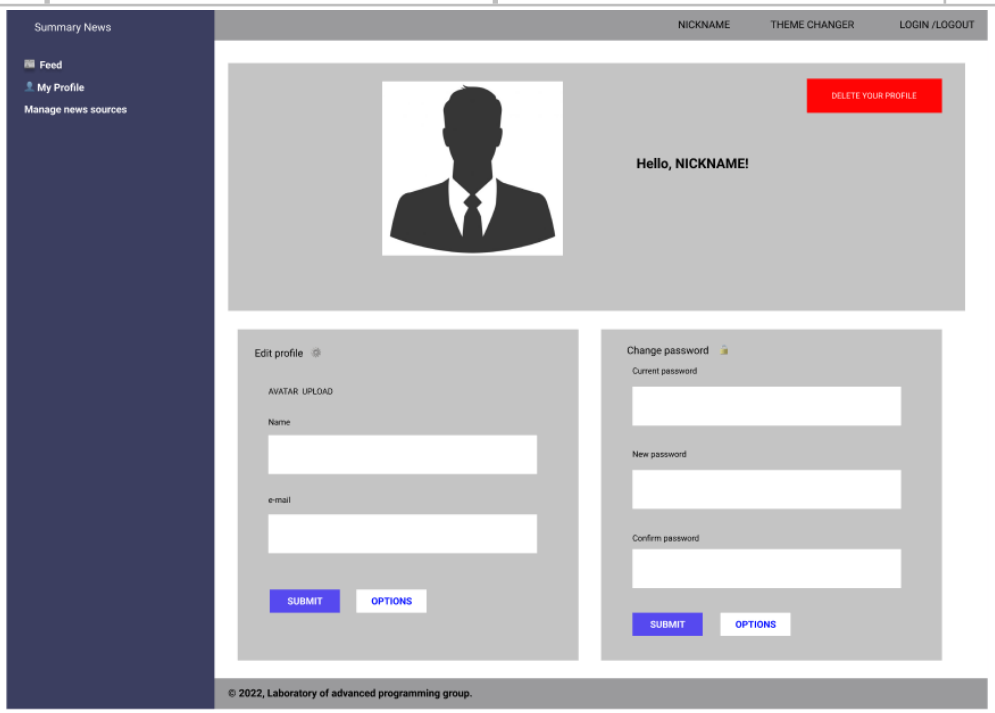
User	Log out from the system	I can log out from the system, using the logout button	20/12/2021
User	Go to the profile page	I can go to the “my profile page”, using a button	16/12/2021



Profile

As a	I want to	So that	Date added
User	Access to my personal info	I can check and edit my personal information or login data	20/12/2021
User	Delete my account	I can delete my personal info from the system, using the delete button	21/12/2021
User	Change theme	I can change the theme's color (light or dark), using the Theme Changer button	19/12/2021
User	Go to the Feed page	I can go to the Feed page, using the Feed button	16/12/2021
User	Log out from the system	I can log out from the system, using the logout button	20/12/2021

User	Go to the manage news sources page	I can go to the manage news sources page, using the manage news sources page button	19/12/2021



News Feed – Booklets

Effort Estimation

Type	Name	Specifications	FP
ILF	**User**	1 RET: User 4 DET: id, email, name, password	LOW (7FP)
ILF	**Rss_Feed**	1 RET: Rss_Feed 4 DET: id, url, rank, user_id	LOW (7FP)
ILF	**Fetched_links**	1 RET: Fetched_links 2 DET: urls	LOW (5FP)
EIF	**News_article**	1 RET: News_article 2 DET: title, text	LOW (5FP)
	Total		24 FP

Transactions

Sign-up with OAuth	EI: 1 FTR: User, 3 DET: user_id, name, email	Low (3FP)
Login with OAuth	EQ: 1 FTR: User, 2 DET: user_id, email	Low (3FP)
Sign-up with standard credentials	EI: 1 FTR: User, 4 DET: user_id, name, email, password	Low (3FP)
Login with standard credentials	EO: 1 FTR: User, 3 DET user_id, email, password	Low (3FP)
Change password	EI: 1 FTR: User, 3 DET:user_id, email, password	Low (3FP)
Delete account	EI: 1 FTR: User, 4 DET: user_id, email, name,	Low

	password	(3FP)
Profile page	EQ: 1 FTR: User, 3 DET: user_id, name, email	Low (3FP)
Insert rss_feed link in database	EI: 2 FTR: User, Rss_Feed, 4 DET: user_id, email, url, rank	Low (3FP)
Update user rss rank	EI: 2 FTR: User, Rss_Feed, 5 DET: user_id, email, url, rank	Medium (4FP)
Fetch the articles in rss_feed link	EI: 3 FTR: Rss_Feed, Fetched_links, User, 4 DET: user_id, email, rss_url, link_urls	Medium (4FP)
Retrieve a news	EQ: 3 FTR: Fetched_link, News_article, User 5 DET: user_id, email, link_urls, news_title, news_text	Low (4FP)
Summarize an article	EO: 2 FTR: User, News_article 3 DET: user_id, email, article_text	Low (4FP)
Extend the text of an article	EQ: 2 FTR: User, News_article 3 DET: user_id, email, article_text	Low (4FP)
Total		**44FP**

In each transaction that involves a protected API call, we have to check that the user email provided with the token is in the database, this is the reason why we need the user id and the email for each call.

Storage functionalities	24 FP
Transactional Functionalities	44 FP
Total	68 FP

Total Function Points: 68 UFP

These are unadjusted function points, so they don't take into account the general characteristics of the system.

We'll take these into account in the effort estimation using COCOMO II

FP TO SLOC

Considering python as a Third Generation language we have an equivalent of **5440 SLOC**

COCOMO II

Using COCOMO II we can compute the effort for the project and estimate the time needed.

We have two models, the early design model, and the post architecture model.

The two models have in common 5 scaling factors

$$E = B + 0.01 * \sum_{j=1}^{n} SF_j$$

B is a constant whose value is 0.91

For the early design model we have 7 Effort Multipliers, so the effort is:

$$PM = A * S^E * \prod_{i=1}^{7} EM_i$$

For the post architecture model we have 17 Effort Multipliers, so the effort is:

$$PM = A * S^E * \prod_{i=1}^{17} EM_i$$

Scale Factors

PREC	NOMINAL	Limited familiarity with past similar works	3.72
FLEX	NOMINAL	Considerable conformance with external interface specifications.	3.04
RESL	LOW	Low risk management needed. Low percent of development schedule/budget devoted to design/risk management.	5.65
TEAM	HIGH	Basically, cooperative interaction between team and stakeholder. High team cohesion.	2.19
PMAT	LOW	Basic CMMI level. Process objectives accomplished but process executed in accordance with low constraining policies.	6.25

Hence

$$E = 0.91 + 0.01 * 20.85 = 1.1185$$

Effort Multipliers – Early Design

For the early design model we have 7 adjusting factors.

$$PM = 2.94 * S^{1.1185} * \prod_{i=1}^{7} EM_i$$

Where the Effort multipliers are:

PERS	HIGH	We have high ACAP, nominal PCAP and very high PCON, since a full analysis of the system has been performed, the team members have nominal programming capability and there is no turnover among the team members.	0.83
RCPX	LOW	We don't need a strong reliability and a complex documentation, and the database size is small, the product complexity is high.	0.83
PDIF	LOW	We had not strict constraints on time, and storage, and the platform volatility is low.	0.87
PREX	NOMINAL	We have nominal platform experience, nominal application experience and a nominal language and tools experience.	1.00
FCIL	VERY HIGH	We use strong and mature software tools, and we meet through video calls.	0.73
RUSE	NOMINAL	High reusability. Services independent from each other so they can be reused across programs.	1.00
SCED	VERY LOW	Low schedule constraints, no schedule stretch-out or acceleration	n/a

Effort Adjustment Factor (EAF) = 0.437

Results – Early Design

Effort

$$PM = 2.94 * 5,44^{1.1185} * 0.437 = 8,54\ person/month$$

Development Time

C=3.67, D=0.28, F=D+0.2*(E-B)=0.32

$$T = C * PM^{F} * SCED = 3.67 * 8,54^{0.32} * 1 = 7,29\ month$$

Cost

Estimating a cost of 2200 $/(person/month)

$$C = PM * 2200 = 18788\ \$$$

Effort Multipliers – Post Architecture

For the post-architecture model we have 17 adjusting factors.

$$PM = 2.94 * S^{1.1185} * \prod_{i=1}^{17} EM_i$$

Where the Effort multipliers are:

Product	RELY	LOW	Low reliability required. The effect of a failure is just the user not being able to read the news.	0.92
Product	DATA	LOW	2 tables and 8 attributes in the database, so low database size.	0.9
Product	CPLX	HIGH	High product complexity. Micro service architecture. ML utilization for processing articles.	1.17
Product	RUSE	HIGH	High reusability. The services are independent from each other so they can be reused across programs	1.11
Product	DOCU	NOMINAL	Right-sized documentation for life-cycle needs.	1.00
System	TIME	NOMINAL	No constraints on resources utilization.	n/a
System	STOR	VERY LOW	No constraints on storage utilization.	n/a
System	PVOL	LOW	Low platform volatility. Major and minor changes are expected to happen every 12 mo	0.87
Personal	ACAP	HIGH	High analyst capability. Ability to define specific goals for project realization.	0.85
Personal	PCAP	NOMINAL	Nominal programming capability. Limited previous experience with functional and non functional requirements.	1.00
Personal	APEX	NOMINAL	Nominal experience with distributed applications architecture.	1.00
Personal	PLEX	NOMINAL	Nominal platform experience.	1.00
Personal	LTEX	NOMINAL	Limited language and tool experience.	1.00
Personal	PCON	VERY HIGH	No turnover while developing. Some team members during the whole period.	0.81
Project	TOOLS	VERY HIGH	Used strong and mature software tools.	0.78
Project	SITE	HIGH	We meet through online meetings, and multi-city development.	0.93
Project	SCED	NOMINAL	Low schedule constraints, no schedule stretch-out or acceleration	1.00

Effort Adjustment Factor (EAF) = 0.45

Results – Post Architecture

Effort

$$PM = 2.94 * 5.44^{1.1185} * 0.45 = 8.8\, person/month$$

Development Time

C=3.67, D=0.28, F=D+0.2*(E-B)=0.32

$$T = C * PM^{F} * SCED = 3.67 * 8.8^{0.32} * 1 = 7.4\, month$$

Cost

Estimating a cost of 2200 $/(person/month)

$$C = PM * 2200 = 19367\, \$$$

4. System Architecture


Web User Interface (Vue.js + NGINX)
REST
API Layer - Gateway (Python 3 + FastAPI)
JSON RPC
Authentication Layer (Python 3)
JSON RPC
JSON RPC
JSON RPC
JSON RPC
Persitance (Python 3)
NLP Analysis (Python 3)
Fetcher (Python 3)
Newspaper (Python 3)
PostgreSQL database engine
Database (Postgres)

5. Sprints Analytics

Each sprint lasts 14 days (2 weeks).

Sprint 1 (15/11/2021 – 28/11/2021)

Task Title	Task Owner	Amount of Work in hours			Start Date	Due Date	Duration
		EST.	COMP.	REM.			
Project Conception and Initiation		42	46	-4			
Proposal of Ideas	EVERYONE	4	4	0	15/11/2021	21/11/2021	7
Research	EVERYONE	8	8	0	22/11/2021	28/11/2021	7
Planning	EVERYONE	10	12	-2	15/11/2021	28/11/2021	14
Scope and goals settings	EVERYONE	6	6	0	22/11/2021	28/11/2021	7
Guidelines	EVERYONE	4	6	-2	22/11/2021	28/11/2021	7
Project Initiation	EVERYONE	10	10	0	22/11/2021	28/11/2021	7

Sprint 2 (29/11/2021 – 12/12/2021)

Task Title	Task Owner	Amount of Work in hours			Start Date	Due Date	Duration
		EST.	COMP.	REM.			
Skeleton of the project		35	37	2			
Make up a template base for the whole app	EVERYONE	10	7	3	29/11/2021	5/12/2021	7
Mockup of landing page	MUSERRA	3	4	-1	29/11/2021	5/12/2021	7
Make a scratch design	SHAIDULLIN	5	5	0	29/11/2021	5/12/2021	7
Setup poetry as tool management for python	IONTA - RAZOVIC	7	8	-1	6/12/2021	12/12/2021	7
Poetry - Configuration fetcher API	IONTA - RAZOVIC	7	9	-2	6/12/2021	12/12/2021	7
Documentation	MUSERRA	3	4	-1	6/12/2021	6/12/2021	1

Sprint 3 (13/12/2021 – 26/12/2021)

Task Title	Task Owner	Amount of Work in hours			Start Date	Due Date	Duration
		EST.	COMP.	REM.			
Back-end setting		37	40	-3			
Test auth using node.js	LAMANNA	9	9	0	13/12/2021	23/12/2021	11
Add a persistence layer	IONTA - RAZOVIC	11	13	-2	13/12/2021	27/12/2021	11
Setup local REST webserver for dev API	IONTA - RAZOVIC	10	10	0	13/12/2021	23/12/2021	15
Fetcher API	MUSERRA	7	8	-1	13/12/2021	22/12/2021	11
Front-end Setting		25	27	-2			
Make Feed Page	SHAIDULLIN-RAZOVIC	5	5	0	13/12/2021	19/12/2021	7
Present feed information	SHAIDULLIN-RAZOVIC	5	6	-1	13/12/2021	19/12/2021	7
Profile page	SHAIDULLIN-RAZOVIC	5	5	0	20/12/2021	26/12/2021	7
Login page	SHAIDULLIN-RAZOVIC	2	2	0	21/12/2021	26/12/2021	6
Sign up page	SHAIDULLIN-RAZOVIC	2	2	0	21/12/2021	26/12/2021	6
Manage news sources page	SHAIDULLIN-RAZOVIC	2	3	-1	22/12/2021	26/12/2021	4
Integrate in docker	SHAIDULLIN-RAZOVIC	2	2	0	27/12/2021	27/12/2021	1
Attached ML API	SHAIDULLIN-RAZOVIC	2	2	0	27/12/2021	27/12/2021	1

Sprint 4 (27/12/2021 – 09/01//2022)

Task Title	Task Owner	Amount of Work in hours			Start Date	Due Date	Duration
		EST.	COMP.	REM.			
Containerization		23	31	-8			
Setup Docker	EVERYONE	10	14	-4	27/12/2021	5/1/2022	10
Build docker img every time a push is performed	RAZOVIC	7	7	0	27/12/2021	30/12/2021	4
Deploy docker image on remote environment	RAZOVIC	3	5	-2	27/12/2021	2/1/2022	7
Integrate front end in docker	RAZOVIC	3	5	-2	3/1/2022	7/1/2022	5

Sprint 5 (10/01/2022 – 23/01/2022)

Task Title	Task Owner	Amount of Work in hours			Start Date	Due Date	Duration
		EST.	COMP.	REM.			
Back-end setting		16	18	-2			
Refine authentication module	LAMANNA	6	6	0	10/01/2022	15/1/2022	6
Exploit authentication module for securing the system	EVERYONE	10	12	-2	10/01/2022	23/1/2022	14

Sprint 6 (24/01/2022 – 06/02/2022)

Task Title	Task Owner	Amount of Work in hours			Start Date	Due Date	Duration
		EST.	COMP.	REM.			
Back-end setting		35	36	-1			
Call protected APIs	IONTA-RAZOVIC-SHAIDULLIN	10	10	0	24/01/2022	6/2/2022	14
Retrieve full article text on-demand	IONTA	5	5	0	24/01/2022	29/01/2022	6
Retrieve full article text on-demand	IONTA	5	5	0	24/01/2022	29/01/2022	6
Endpoint for logout	IONTA	2	2	0	31/01/2022	31/01/2022	1

Endpoint for getting all user persisted info	IONTA	2	2	0	01/02/2022	01/02/2022	1
Endpoint for adding a RSS Feed to a certain user	IONTA	2	2	0	02/02/2022	02/02/2022	1
Endpoint for deleting a user	IONTA	2	2	0	03/02/2022	03/02/2022	1
Endpoint signatures refactor	IONTA	2	2	0	04/02/2022	04/02/2022	1
Endpoint for user sign-in from form data	IONTA	1	2	-1	04/02/2022	05/02/2022	1
Add authentication using email/password	LAMANNA - RAZOVIC - SHAIDULLIN	4	4	0	01/02/2022	04/02/2022	4
Front-end Setting		29	42	-11			
Connect to backend by means of endpoints	SHAIDULLIN-RAZOVIC	4	6	-2	05/02/2022	6/2/2022	2
Pop-up window for triggering Social Login	SHAIDULLIN-RAZOVIC	4	6	-2	24/01/2022	26/1/2022	2
Check User Authentication on the basis of Access Token	SHAIDULLIN-RAZOVIC	5	8	-3	26/01/2022	31/1/2022	5
Give the user the possibility to enter a RSS URL	SHAIDULLIN-RAZOVIC	3	3	0	05/02/2022	6/2/2022	2
Call protected APIs	SHAIDULLIN-RAZOVIC	3	3	0	24/01/2022	26/1/2022	2
Display full text of an article	SHAIDULLIN-RAZOVIC	1	1	0	26/01/2022	31/1/2022	5
Attach delete user functionality	SHAIDULLIN-RAZOVIC	2	3	1	01/02/2022	3/2/2022	2
Cleaning wired things	SHAIDULLIN-RAZOVIC	2	4	-2	02/02/2022	4/2/2022	2
Attach change password Functionality	SHAIDULLIN-RAZOVIC	1	1	0	04/02/2022	05/02/2022	1
Switch RSS feed	SHAIDULLIN-RAZOVIC	2	2	0	03/02/2022	5/2/2022	2

Burndown Data

Sprint 1

Day	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Plan	229	226	224	221	218	215	213	210	207	204	202	199	196	194
Estimate	229	226	223	221	219	217	215	211	207	202	197	193	190	186
Hrs Completed	3	3	2	2	2	2	4	4	5	5	4	3	4	2
Hrs Remaining	226	223	221	219	217	215	211	207	202	197	193	190	186	184

Sprint 2

Day	15	16	17	18	19	20	21	22	23	24	25	26	27	28
Plan	191	188	185	183	180	177	174	172	169	166	164	161	158	155
Estimate	184	181	178	176	173	171	168	165	163	160	157	155	152	149
Hrs Completed	3	3	2	3	2	3	3	2	3	3	2	3	3	2
Hrs Remaining	181	178	176	173	171	168	165	163	160	157	155	152	149	147

Sprint 3

Day	29	30	31	32	33	34	35	36	37	38	39	40	41	42
Plan	153	150	147	144	142	139	136	134	131	128	125	123	120	117
Estimate	147	142	136	130	126	120	116	109	104	97	91	86	80	80
Hrs Completed	5	6	6	4	6	4	7	5	7	6	5	6	0	0
Hrs Remaining	142	136	130	126	120	116	109	104	97	91	86	80	80	80

Sprint 4

Day	43	44	45	46	47	48	49	50	51	52	53	54	55	56
Plan	115	112	109	106	104	101	98	95	93	90	87	85	82	79
Estimate	80	78	75	73	70	68	68	65	62	59	56	56	53	50

Hrs Completed	2	3	2	3	2	0	3	3	3	3	0	3	3	2
Hrs Remaining	78	75	73	70	68	68	65	62	59	56	56	53	50	48

Sprint 5

Day	57	58	59	60	61	62	63	64	65	66	67	68	69	70
Plan	76	74	71	68	65	63	60	57	55	52	49	46	44	41
Estimate	48	47	45	43	42	40	38	37	36	35	34	33	32	31
Hrs Completed	1	2	2	1	2	2	1	1	1	1	1	1	1	1
Hrs Remaining	47	45	43	42	40	38	37	36	35	34	33	32	31	30

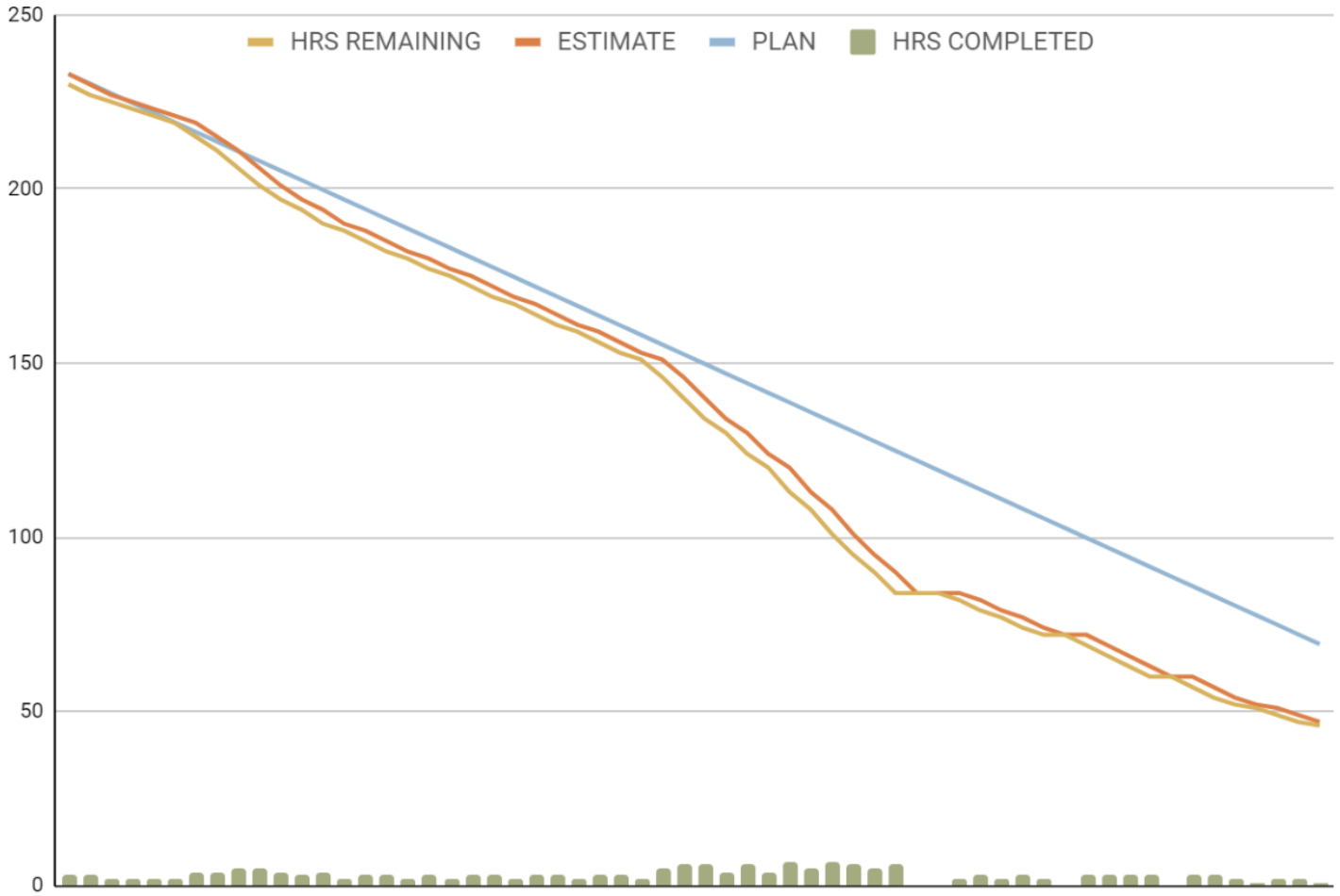
Sprint 6

Day	71	72	73	74	75	76	77	78	79	80	81	82	83	84
Plan	39	38	35	32	30	25	22	15	12	9	6	4	1	-2
Estimate	34	30	25	21	16	11	8	1	-5	-11	-18	-22	-26	-32
Hrs Completed	4	5	4	5	5	3	7	6	6	7	4	4	6	4
Hrs Remaining	30	25	21	16	11	8	1	-5	-11	-18	-22	-26	-32	-36

Total

Total Estimated Hours	233
Completed Hours	269
Remaining Hours	-36
Total Days	84
Average Hours per Day	2.72

Chart


HRS REMAINING
ESTIMATE
PLAN
HRS COMPLETED
250
200
150
100
50
0

6. Project Repository