

H A N A D i g i t a l P o w e r O N P r o j e c t

하나 디지털 파워 온 프로젝트

Project Name : 불필요한 소비 줄이기를 도와줄 두꺼비 서비스

Team Name : ONLY ONE (김건호, 김수민, 윤찬익, 이승범, 임경민)



CONTENT

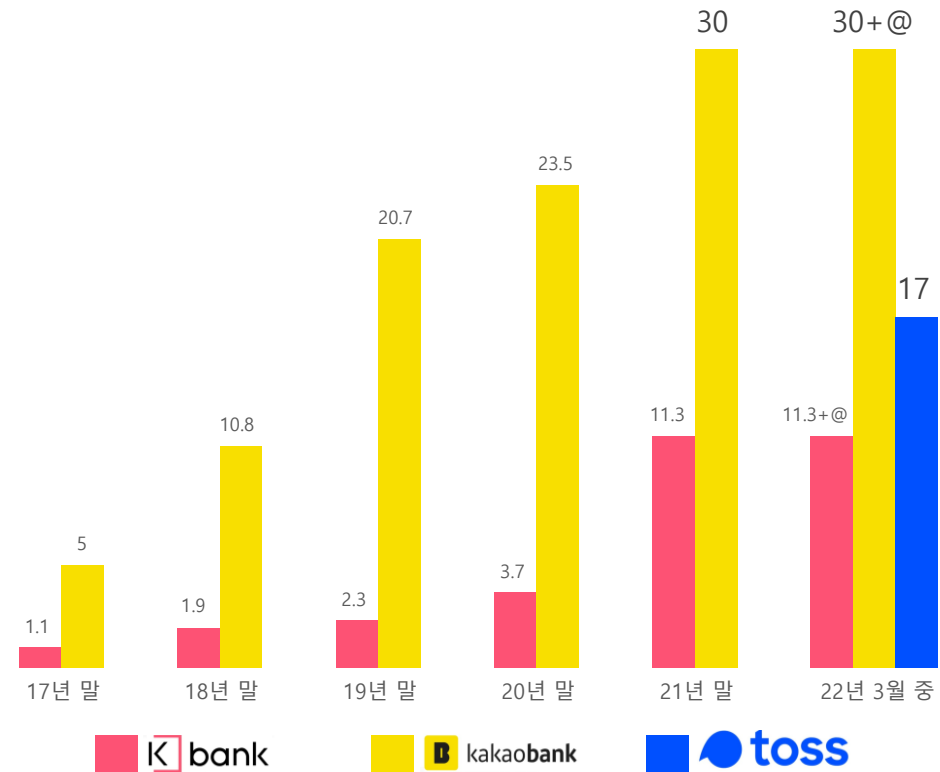
- 01. Proposal Background
저희 서비스 아이디어의 제안 배경을 담았습니다.
- 02. Introduce
본격적인 저희 서비스 아이디어의 소개입니다.
- 03. Technical Application
서비스 아이디어에 적용된 기술을 보여드리겠습니다.
- 04. End _ benefit
저희 서비스의 기대효과로 마무리하겠습니다.

Targeting _ MZ세대

- 최근 금융권의 경계가 무너지고 있다는 점은 은행들이 MZ세대 마케팅에 공을 들이는 이유 중 하나입니다. 특히 첨단 디지털 기술에 기반한 혁신적인 서비스로 무장한 시중은행에게 빅테크는 가장 큰 경쟁자입니다.
- 금융권 관계자는 "카카오뱅크와 토스의 급성장은 MZ세대가 금융시장에 미치는 파급력을 보여주는 대표적인 사례"라며 "편리함이라는 새로운 무기를 앞세워 등장한 후 MZ세대들이 직접 사용하면서 입소문을 통해 성장했고 그 입소문은 부모세대까지 고객으로 끌어들이었다"라고 평가했습니다.
- 은행들은 앞으로도 이런 추세가 계속될 것으로 보고 있습니다. 오픈뱅킹과 마이데이터 산업, 빅테크 기업의 등장으로 금융서비스를 누릴 수 있는 창구는 더 많아졌고, 그런만큼 MZ세대의 영향력도 계속 유지될 것 이란 전망입니다.

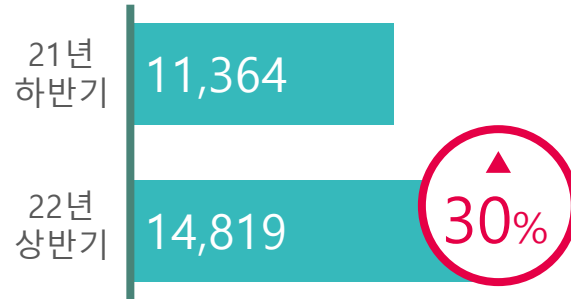
자료 : Biz Watch, 이경남 기자, 은행들은 왜 MZ세대에 열광할까

인터넷전문은행 3사 수신잔액 추이(단위 : 조원)

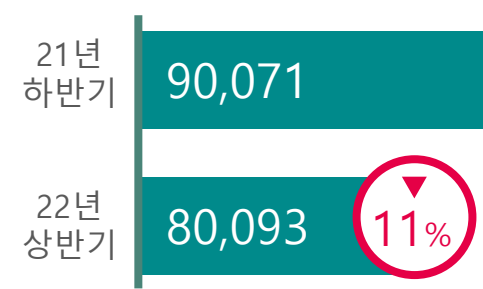


자료 : 은행연합회 및 각사

무지출·무소비 언급량



플렉스·올로 언급량



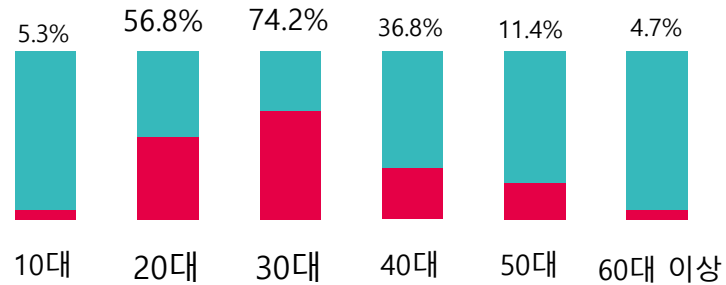
자료 : KPR 디지털커뮤니케이션연구소

'영끌'과 '빚투'가 할퀴고 간 자리에 고물가까지 이어지면서 최근 M세대는 허리띠를 졸라매고 있습니다. 몇 해전까지만 해도 '올로(YOLO)'나 '플렉스(FLEX)' 같은 과시적 소비가 유행했으나, 이제는 한 푼이라도 아끼기 위한 '짠테크' 열풍이 불고 있는 모습입니다. 고물가·고금리에 불경기까지 겹치는 등 경제적 상황이 악화하자 미래를 대비하고자 하는 이들이 늘어난 셈입니다.

자료 : 아시아경제, 허미담 기자, '영끌'에서 '짠테크'로...저성장 시대 고착화

짠 + 재테크 = 짠테크

짠테크에 관심이 많은 연령대는?



자료 : 푸본현대생명

Targeting
Point

짠테크 관련 전반적인 인식 평가(단위 : %)

코로나19 이후 짠테크에 관심을 갖는 사람들이 많아졌다.

72.6

요즘처럼 불황이 이어지는 시기에 지출을 관리하는 것이 중요하다.

94.5

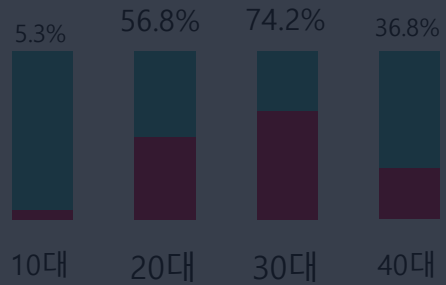
요즘처럼 불황이 이어지는 시기에 소액이라도 저축을 해야 한다.

89.0

자료 : 엠브레인 트렌드모니터



짬테크에 관심이 많은 연령대는?



Targeting Point



짬테크 관련 전반적인 인식 평가(단위 : %)

코로나19 이후 짬테크에 관심이 많은 사람들이 많아졌다.

72.6

이 이어지는 시
하는 것이 중

94.5

어지는 시
축을 해야

89.0

드모니터

불필요한 소비를 줄여주는 도우미, 두꺼비



Background

Introduce

Application

End



[기존 서비스]

- 연령을 기준으로 소비 항목별 소비현황을 비교
- 이체 및 출금 부분에서 소비를 항목화하기 어려움
- 재무구조 변화에 반응하지 않음.

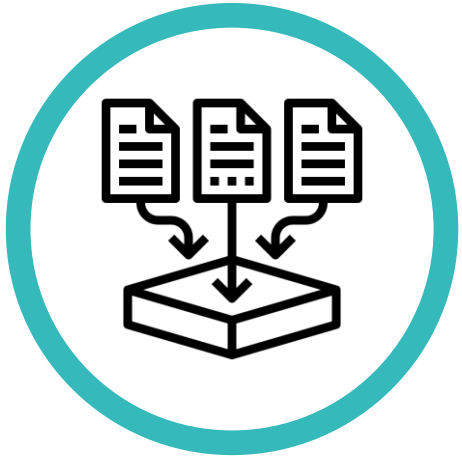


[두꺼비 서비스]

- 재무구조를 기준으로 소비 항목별 소비현황을 비교
- 이체 및 출금 부분에서 소비항목화 가능
- 재무구조 변화에 반응하여 소비항목 별로 절약 추천



저희가 제안하는 아이디어를 현실화하기 위해 사용된 IT기술입니다. 마이데이터를 통해 소비내역을 파악하고, AI를 바탕으로 소비패턴을 파악합니다.



01. MY Data

마이데이터를 활용하여 사용자의 소비내역을 확보합니다. 체크카드, 신용카드를 포함하여 이체내역 또한 유의미한 소비내역으로 활용하기 위해 이체 후 간단한 항목을 제시하여 이용자가 직접 분류할 수 있도록 합니다.



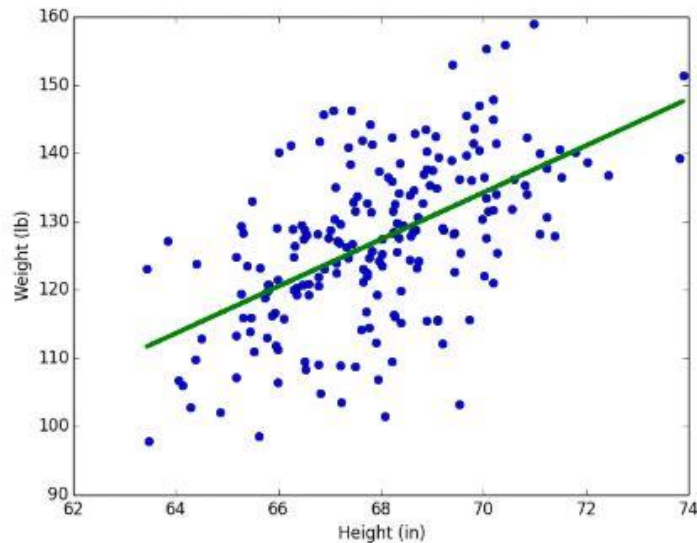
02. AI

확보된 마이데이터를 바탕으로 이용자의 현 재정상황을 파악하여 알고리즘을 활용하여 소비내역치를 조정합니다. 이때 사용되는 알고리즘은 예측 알고리즘으로, 수많은 과거 데이터를 기반으로 예측치를 측정합니다.



02-1. AI 구현 계획

재정상황을 독립변수로, 소비내역을 종속변수로 두고 예측 알고리즘을 통해 수 많은 과거 데이터를 기반으로 예측치를 측정합니다. 저희는 선형회귀를 활용하여 독립변수와 종속변수의 관계성을 도출했습니다.



02-2. 선형회귀

선형회귀는 종속 변수와 한 개 이상의 독립 변수 간의 선형 관계를 모델링하는 회귀 분석으로, 주어진 독립 변수와 종속 변수 사이의 관계를 가장 잘 설명하는 선형 함수를 찾습니다.

선형회귀는

- 데이터 관계 모델링
 - 독립 변수의 계수를 통해 변수의 중요도를 평가
 - 각 변수가 종속 변수에 어떤 영향을 미치는지 파악
 - 독립 변수의 값을 기반으로 종속 변수의 값을 예측
- 위 4가지 기능을 수행합니다.

02-3. AI 구현 _ 데이터 확보

학습을 위한 임시의 데이터를 준비합니다. 저희가 준비한 데이터는 마이 데이터를 통해 확보한 데이터로 가정하고 10000개의 데이터셋을 준비했습니다.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
user_id	식료품	의류	가전제품	스포츠용품	교통	미용	의료	여행	문화	음식점	주류	통신	가정용품	도서	유류비	반려동물	유아용품	교육	현재 소득	현재 지출	저축	금융 자산	부채 금액	재무적 안정도	대출 상황
1	1298	366	3943	419	2	107	32	146	98	374	2858	93	319	187	1212	113	132	4327	32546	17677	22403	1630	15892	7	27
2	3277	51	4408	215	79	18	1247	192	470	2615	88	78	465	107	1620	0	139	2507	59031	62753	11256	47619	39213	5	94
3	2249	2845	8912	65	115	170	653	81	299	1603	1137	34	173	149	1423	188	10	259	50983	49765	26514	13648	544	3	53
4	577	1579	9463	52	189	385	925	85	220	1090	348	94	436	48	1074	97	157	1519	36856	24172	5967	77531	13981	7	66
5	3122	2686	7611	409	410	454	1735	607	456	1204	780	44	595	132	2431	137	192	3627	49624	15025	965	55853	26663	6	29
6	2662	2953	8603	423	777	352	35	308	424	695	1102	78	593	52	66	149	87	3376	22305	56168	13071	94340	20497	10	17
7	2655	1337	8701	116	12	66	1991	166	413	2437	744	40	333	89	1579	65	94	4083	32942	64268	5728	88029	47664	4	80
8	773	1455	1193	39	728	129	724	528	87	1104	2426	36	474	155	1071	110	54	3204	34969	29833	27442	10153	30033	10	68
9	2375	1415	2496	480	799	310	2376	267	191	430	862	50	493	25	2935	132	104	4484	44332	57078	10763	95038	38963	8	26
10	4396	668	6406	351	584	441	2814	745	297	788	1607	47	365	69	1522	91	92	1816	33002	63827	16900	80867	14341	3	40
11	3314	753	4959	194	930	483	197	608	141	2474	1203	49	5	53	1467	128	78	1128	52398	15855	10844	42607	43482	0	94
12	3345	2355	1881	175	682	356	690	385	351	2212	1080	46	384	147	242	33	161	382	23767	50063	16314	81702	17441	0	64
13	4606	1438	3109	367	235	183	831	101	399	1816	1758	70	29	36	2063	121	25	9	57453	20958	21543	31583	35139	9	95
14	2420	266	7103	350	215	347	2682	838	466	1293	509	17	171	115	2597	68	140	3707	45995	15517	26711	79585	7199	10	75
15	2709	2227	159	160	419	364	589	866	356	2343	1234	18	403	24	204	116	193	2687	30195	22183	9369	86794	24729	0	17
16	4581	2664	6954	399	699	476	933	458	496	1473	831	26	595	189	1350	36	88	1649	24844	39874	14969	7969	4278	2	3
17	2084	604	8814	362	178	331	1735	455	312	2789	1115	91	72	173	1221	183	143	393	59964	32704	19383	94823	36686	1	2
18	2821	1088	6551	127	195	121	2152	678	291	2099	2386	99	464	150	99	192	200	542	23121	53217	18304	64867	11100	5	35
19	4282	866	632	89	265	283	190	757	91	1940	2608	36	112	34	0	110	138	114	52888	65794	10593	10414	39521	5	90
20	2936	2687	2242	35	61	392	2160	247	322	2741	748	4	261	120	2404	192	149	782	25627	17332	14628	8706	2358	3	40
21	49	2827	520	487	673	385	1075	52	241	390	1498	64	579	94	2945	132	191	2376	40101	20034	16951	24792	44714	2	80
22	3249	1403	6098	8	214	408	283	806	258	1716	472	91	285	95	1568	91	5	3551	56406	62625	20580	17765	28088	6	59
23	4804	833	9073	325	942	267	2242	279	265	616	2021	91	466	48	184	127	127	127	127	127	127	127	127	2	67
24	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015	1015

```
# 소비 카테고리 및 재정 상황 카테고리
expense_categories = list(categories.keys())
financial_categories = list(financial_categories.keys())

# 데이터셋 로드
dataset = []
with open(dataset_file, 'r', encoding='cp949', errors='ignore') as csvfile:
    reader = csv.DictReader(csvfile)
    for row in reader:
        dataset.append(row)

# 입력과 출력 데이터 준비
X = []
Y_expense = []
Y_savings = []

for data in dataset:
    x = []
    for f_category in financial_categories:
        x.append(float(data[f_category]))
    X.append(x)

    y_expense = []
    y_savings = []
    for category in expense_categories:
        y_expense.append(float(data[category]))
        y_savings.append(float(data['저축']))
    Y_expense.append(y_expense)
    Y_savings.append(y_savings)
```

02-4. AI 구현 _ 알고리즘 구현

10000개의 데이터셋을 학습시킬 알고리즘을 다음과 같이 구현했습니다.

```
# 선형 회귀 모델 훈련
model_expense = LinearRegression()
model_expense.fit(X, Y_expense)

model_savings = LinearRegression()
model_savings.fit(X, Y_savings)

# 재정 상황에 따른 소비와 저축 추천 함수
def recommend_expense_and_savings(financial_situation):
    x = []
    for f_category in financial_categories:
        x.append(float(financial_situation[f_category]))
```

02-5. AI 구현 _ 테스트 진행 (1)

재정 상황이 줄어들었을 때를 고려하여 소득치를 줄였을 경우로 테스트를 진행하였습니다.

```
# 현재 재정 상황 입력
current_financial_situation = {
    '현재 소득': 30000,
    '현재 지출': 20000,
    '저축': 5000,
    '금융 자산': 50000,
    '부채 금액': 20000,
    '재무적 안정성': 8,
    '대출 상환능력': 70,
    '신용 점수': 700,
    '투자 수익률': 5,
    '부동산 보유량': 3,
}
```

< 현재 재정 상황 >

```
# 줄어든 재정 상황 입력
reduced_financial_situation = {
    '현재 소득': 20000,
    '현재 지출': 20000,
    '저축': 5000,
    '금융 자산': 50000,
    '부채 금액': 20000,
    '재무적 안정성': 8,
    '대출 상환능력': 70,
    '신용 점수': 700,
    '투자 수익률': 5,
    '부동산 보유량': 3,
}
```

< 줄어든 재정 상황 >

현재 소득치 30,000에서 줄어든 소득치 20,000으로 변화.

02-6. AI 구현 _ 테스트 진행 (2)

소득치가 10,000이 줄어들 때, 현재 소비내역을 바탕으로 재정 상황이 줄어들었을 때의 예측 소비 내역을 결과값을 도출합니다. 예측 소비 내역은 결국 추천 소비 내역으로 인식할 수 있습니다. 실제 데이터를 기반으로 한다면, 더욱 확실한 변화를 보일 것으로 예상합니다. 현재는 정규분포를 가진 랜덤한 데이터셋을 이용하여 예측치가 특정 카테고리에서는 큰 변화를 보이지 않는 점이 아쉬움으로 남았습니다.

```
# 현재 소비 내역 입력
current_expense = {
    '식료품': 2434,
    '의류': 1516,
    '가전제품': 5078,
    '스포츠용품': 252,
    '교통': 503,
    '미용': 252,
    '의료': 1529,
    '여행': 489,
    '문화': 247,
    '음식점': 1479,
    '주류': 1516,
    '통신': 49,
    '가정용품': 301,
    '도서': 97,
    '유류비': 1492,
    '반려동물': 99,
    '유아용품': 100,
    '교육': 2447,
}
```

< 현재 소비 내역 >

소비 카테고리 변화:

식료품:	2434	->	2212
의류:	1516	->	1371
가전제품:	5078	->	4764
스포츠용품:	252	->	213
교통:	503	->	496
미용:	252	->	239
의료:	1529	->	1514
여행:	489	->	364
문화:	247	->	247
음식점:	1479	->	1368
주류:	1516	->	1489
통신:	49	->	48
가정용품:	301	->	292
도서:	97	->	95
유류비:	1492	->	1480
반려동물:	99	->	91
유아용품:	100	->	97
교육:	2447	->	2435

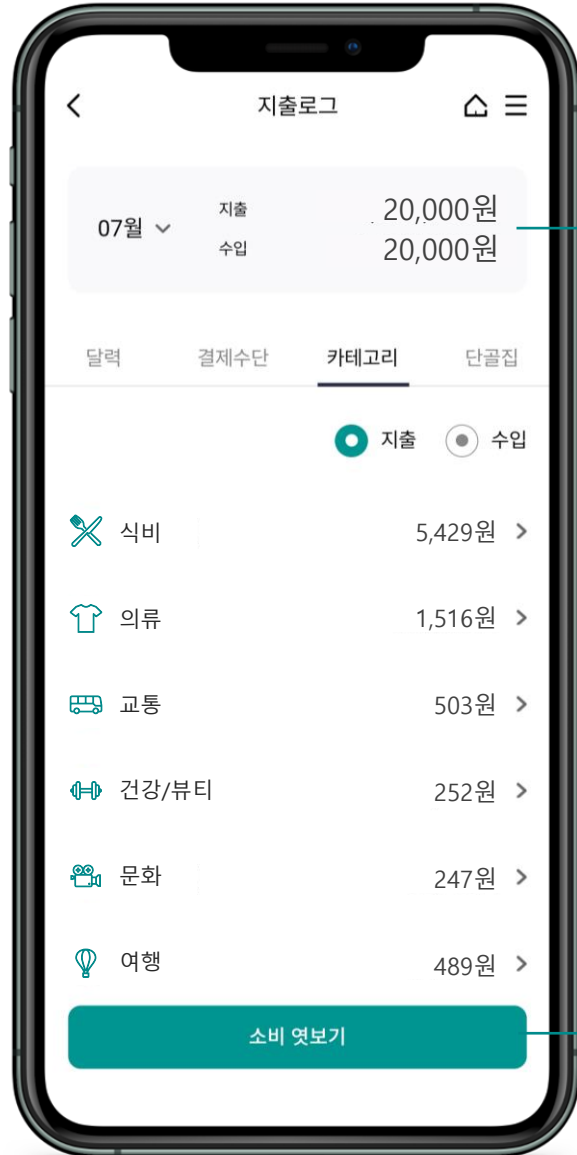
< 예측 소비 내역 >

Background

Introduce

Application

End

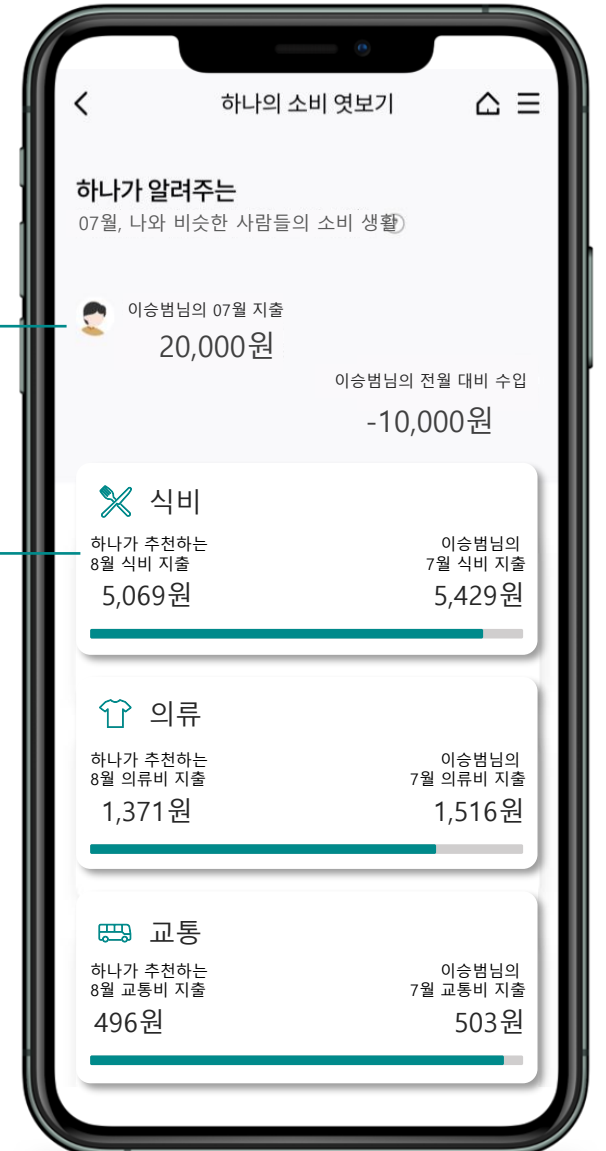


월말에 총지출과 총수입을 기록합니다.
총지출 내에서 소비 카테고리별로 분류하여 소비항목을 나눕니다. 각 항목에 해당하는 금액의 합을 아래 카테고리 탭 밑으로 보여줍니다.

지난 달 수입의 증감을 확인하며 이용자의 재무상태의 변화를 알려줍니다.

수입의 증감을 반영하여 변화된 재무 구조에 맞는 항목별 소비액을 추천합니다.

기존 소비 엿보기를 활용합니다.

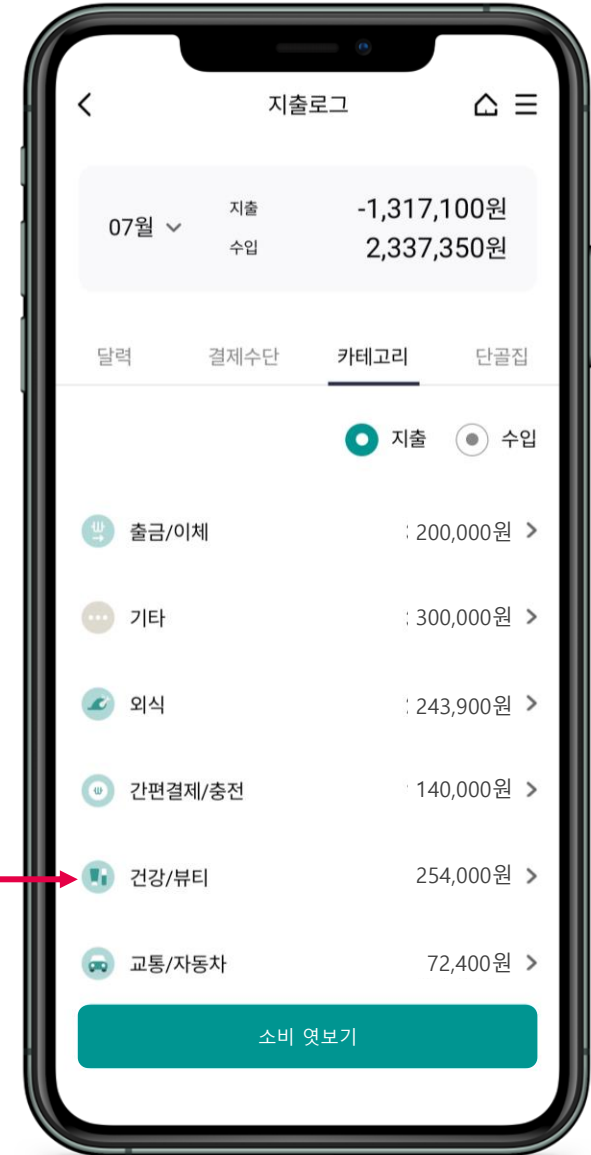
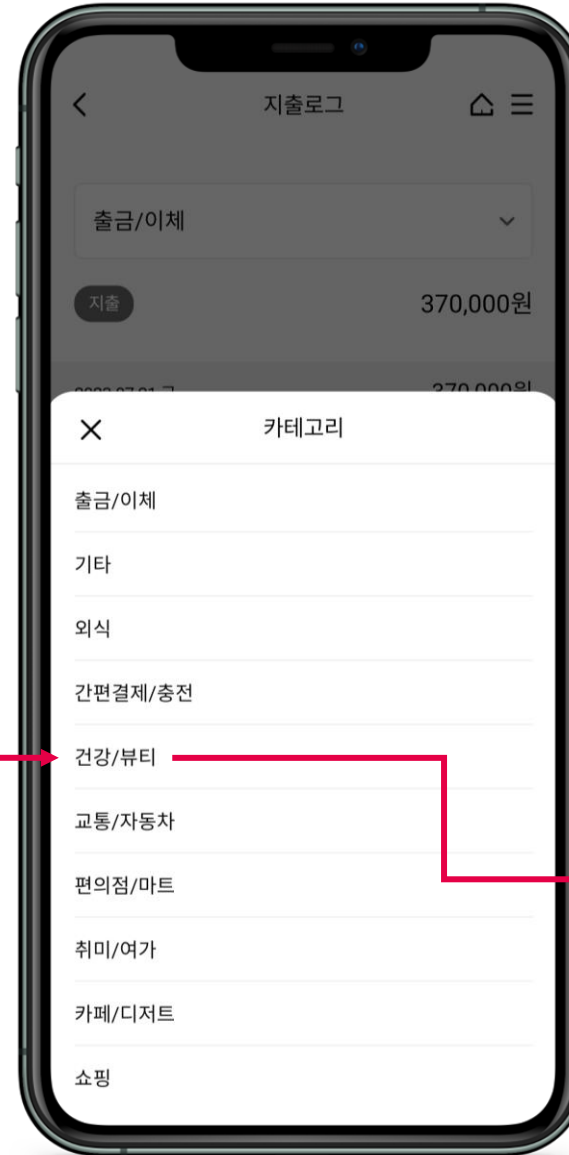
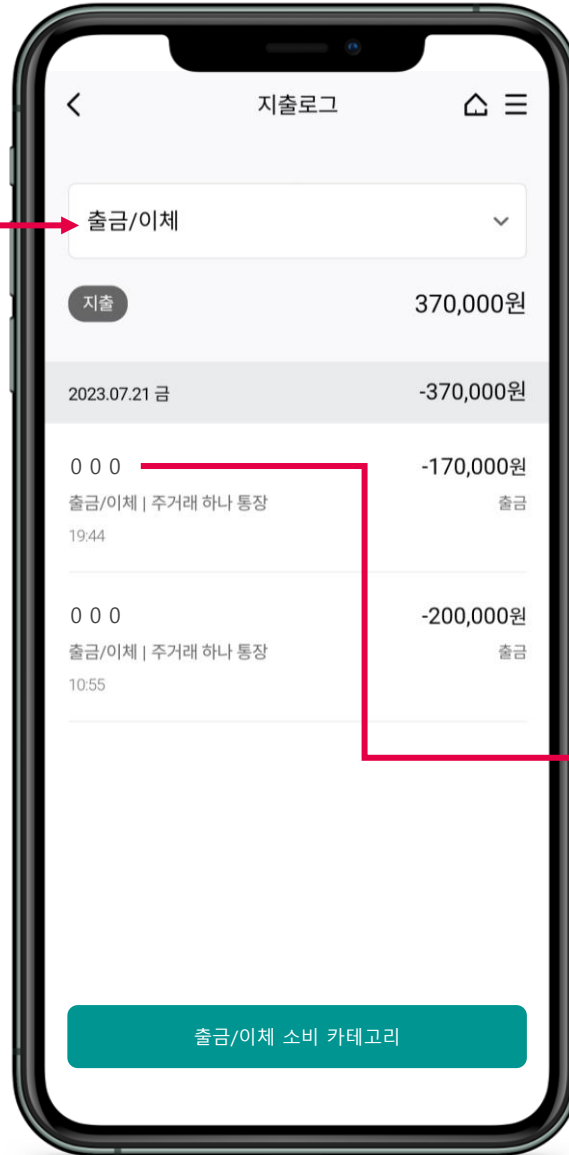
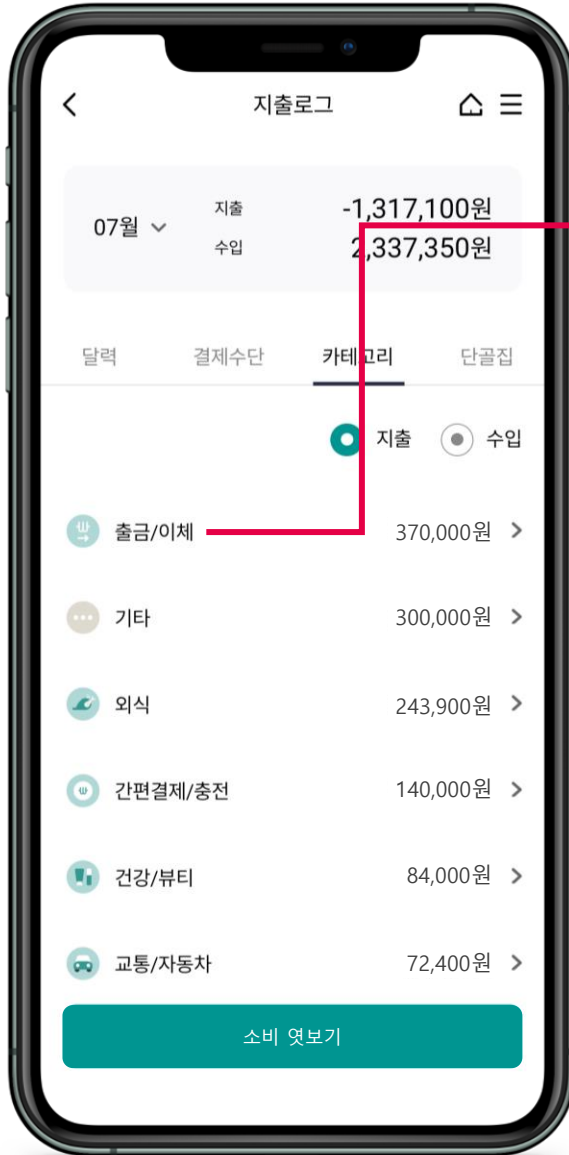


Background

Introduce

Application

End





Strengths

- MZ 고객 확대 및 잠금효과
- 고객의 데이터 확보
- 다양한 금융상품과 결합



Weakness

- 재무구조가 개인의 성향을 반영하기 어려움
- 개인정보동의 필요



Opportunities

- 절약, 짠테크에 관심 많은 MZ 고객 증가
- 은행 앱 사용자 수 증가



Threats

- 타은행의 해당 서비스 도입
- 인터넷 은행의 점유율 증가

