

## SYSTEM REQUIREMENTS

In this chart we list our app may include and we rank them using a hierarchical system called the MoSCoW method. MoSCoW stands for Must have, Should have, Could have, Would have and it is used to determine what elements of the app absolutely need to be included in the initial prototype, and what are not relevant enough to include in the first version.

Requirement	Rationale	Priority	Content	System Function
<b>In-App Calculator</b>	User can manually input the numbers for their expenses, and the app will automatically calculate how much they have left to spend. This would be for cash transactions.	<b>Should have</b>	<ul style="list-style-type: none"> <li>- User can track the amount they they have vs. how much they spent</li> <li>- Can see whether they are above or below budget</li> </ul>	Populates the database with the users' spending habits; numbers can be edited.
<b>Connect to Bank Account</b>	App will automatically see how much the user spent, without them having to input the numbers.	<b>Should have</b>	<ul style="list-style-type: none"> <li>- User can track the amount they they have vs. how much they spent</li> <li>- Can see whether they are above or below budget</li> </ul>	Populates the database with the users' spending habits; numbers cannot be edited.
<b>Exchange Rate Tracking</b>	App will know the exchange currency rates for whatever country they are in, and the app will automatically show the conversion.	<b>Must have</b>	<ul style="list-style-type: none"> <li>- Currency exchange rate by country currency and home-town currency</li> </ul>	Real-time currency exchange rates for whenever the transaction occurred.
<b>User Profiles</b>	User profiles that will allow a personal profile so someone can go back on their trips and see how much they spend	<b>Must have</b>	<ul style="list-style-type: none"> <li>- Users' personal information collected during the onboarding process; can add their banking information</li> </ul>	Allows editing of previously entered information; may be able to share their profile with other people to be on the same budget
<b>Connect to a Restaurant's Price Range</b>	Using Google or Trip Advisor's price range system, this information will be viewable for the user.	<b>Could have</b>	<ul style="list-style-type: none"> <li>- When the user determines their daily price range, the app will show nearby restaurants that will work in their budget</li> </ul>	Connects to Google Maps or Trip Advisor's restaurant reviews and price range

<b>Onboarding</b>	The onboarding will allow the user to start a new budget, and teach them how to put transaction information into the system	<b>Must have // Should have</b>	- User inputs where they are going, how much money they have, and what price range they want to spend	Tutorial that user's can go back to in the "Help" menu.
<b>Filters &amp; Subgroups</b>	There will be predefined spending subgroups that the app already has, as well as the user is able to create their own. When they add a transaction, the user chooses which subgroup the transaction is categorized in.	<b>Must have</b>	- Shows user where they are spending in certain subgroups (food, drinks, activities)	Colour-coded groups for ease to the user when looking at the data.
<b>Notifications</b>	Will tell users how much money that can spend daily; letting users know if they went over their budget	<b>Must have</b>	- Allows the user to have a reminder that they should not spend more than the amount given  - Easy way for the user to be reminded how much money they have left without going in the app	Notifications enabled.
<b>Optical Character Recognition</b>	App will be able to locate numbers when a picture is taken and automatically add to the budget	<b>Should have</b>	- A camera button to add transaction through the camera	Camera access enabled.  Need to be able to take a clear photo where the app can distinguish the type.
<b>Mapping/ Location Services</b>	Application can see where the user is around the world and automatically finds the currency that the area uses; Will also work for nearby restaurants and activities based on the chosen price range that the user distinguishes	<b>Could have</b>	- Map of tourist destinations nearby that the user can see  - Good for the user to see where they have been and where they spent the money	Location services enabled.
<b>Dashboard</b>	At a glance, the user is able to see how much remaining balance they have for their daily budget.	<b>Must have</b>	- Budget balance  - Money already spent that day	Calculator and access to bank account

## ADDITIONS

Requirement	Rationale	Priority	Content	System Function
<b>API Development</b>	The system can go through emails and add transaction from email receipts. Includes place to access all receipt easily, would work well if users did not want to connect bank account information.	<b>Could have</b>	<ul style="list-style-type: none"> <li>- Organizes email receipt for easy access.</li> <li>- Automatically adds all recorded transactions to budget records.</li> </ul>	Access to email.
<b>Search Hub</b>	User can search for a specific location and they will receive information on average costs.	<b>Should have</b>	<ul style="list-style-type: none"> <li>- Food/Living costs</li> <li>- Daily spending estimates based on dates and location.</li> </ul>	<ul style="list-style-type: none"> <li>- Finding data online</li> <li>-Records users spending and adding data to create more accurate data base of spending habits.</li> </ul>

## SUBTRACTIONS

- Connect to a restaurants price range.